

日本国特許庁  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日      2003年 8月15日  
Date of Application:

出願番号      特願2003-293804  
Application Number:

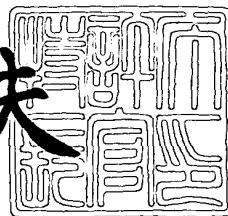
[ST. 10/C] : [JP2003-293804]

出願人      株式会社ソニー・コンピュータエンタテインメント  
Applicant(s):

2003年 8月28日

特許庁長官  
Commissioner,  
Japan Patent Office

今井康夫



出証番号 出証特2003-3070405

【書類名】 特許願  
【整理番号】 SCEI02009A  
【提出日】 平成15年 8月15日  
【あて先】 特許庁長官殿  
【国際特許分類】 G06T 15/00  
G06F 15/00

【発明者】

【住所又は居所】 東京都港区南青山2丁目6番21号 株式会社ソニー・コンピュータエンタテインメント内

【氏名】 大場 章男

【特許出願人】

【識別番号】 395015319  
【氏名又は名称】 株式会社ソニー・コンピュータエンタテインメント

【代理人】

【識別番号】 100105924  
【弁理士】

【氏名又は名称】 森下 賢樹  
【電話番号】 03-3461-3687

【先の出願に基づく優先権主張】

【出願番号】 特願2002-262170  
【出願日】 平成14年 9月 6日

【先の出願に基づく優先権主張】

【出願番号】 特願2003-124477  
【出願日】 平成15年 4月 28日

【手数料の表示】

【予納台帳番号】 091329  
【納付金額】 21,000円

【提出物件の目録】

【物件名】 特許請求の範囲 1  
【物件名】 明細書 1  
【物件名】 図面 1  
【物件名】 要約書 1  
【包括委任状番号】 0211041

**【書類名】特許請求の範囲****【請求項 1】**

入力された3次元オブジェクトのデータをグループに分類する分類部と、

同一グループに属する3次元オブジェクトを包含する部分空間をそれぞれ独立した描画単位とする個別の描画処理を行い、前記描画単位ごとに独立した画像データを生成する描画処理部と、

前記描画単位ごとに生成された前記画像データを統合して最終的に表示すべき画像データを生成する統合部とを含むことを特徴とする画像処理装置。

**【請求項 2】**

前記描画処理部は、透視投影で規定される四角錐台を前記部分空間として算出して前記3次元オブジェクトの透視変換を行う座標変換部と、前記四角錐台における個別の描画処理を行うレンダリング部とを含むことを特徴とする請求項1に記載の画像処理装置。

**【請求項 3】**

前記分類部は、前記3次元オブジェクトの運動特性に関する属性にもとづいて前記3次元オブジェクトをグループ分けすることを特徴とする請求項1または2に記載の画像処理装置。

**【請求項 4】**

前記分類部は、前記3次元オブジェクトを描画する際の詳細度に関する情報にもとづいて前記3次元オブジェクトをグループ分けすることを特徴とする請求項1または2に記載の画像処理装置。

**【請求項 5】**

前記統合部は、前記描画単位ごとに生成された前記画像データの各画素の奥行き値のダイナミックレンジを補正した上で、補正された前記奥行き値の大小比較により前記画像データを合成して最終的に表示すべき画像を生成することを特徴とする請求項1から4のいずれかに記載の画像処理装置。

**【請求項 6】**

前記描画処理部は、複数のレンダリング部を含み、各描画単位の描画処理の複雑度にもとづいて描画処理を前記複数のレンダリング部に分散させることを特徴とする請求項1から5のいずれかに記載の画像処理装置。

**【請求項 7】**

前記描画処理部は、複数の処理性能の異なるレンダリング部を含み、各描画単位の描画処理の複雑度に応じた処理性能をもつ前記レンダリング部にその描画処理を行わせることを特徴とする請求項1から5のいずれかに記載の画像処理装置。

**【請求項 8】**

当該装置の外部にあって、ネットワークで当該装置に結合した分散描画処理ユニットから前記描画単位で描画処理された画像データを受信する通信部をさらに含み、前記統合部は、前記分散描画処理ユニットから受信した画像データを前記描画処理部が生成した画像データに統合し、最終的に表示すべき画像データを生成することを特徴とする請求項1から5のいずれかに記載の画像処理装置。

**【請求項 9】**

各描画単位の描画処理の詳細度に関する情報に応じてネットワーク上の距離の異なる分散描画処理ユニットにその描画処理を行わせることを特徴とする請求項8に記載の画像処理装置。

**【請求項 10】**

ネットワークを介して相互に情報を交換しながら、分散描画処理を行う複数の画像処理装置を含む画像処理システムであって、

入力された3次元オブジェクトのデータをグループに分類する分類部と、

同一グループに属する3次元オブジェクトを包含する部分空間をそれぞれ独立した描画単位とする個別の描画処理を行い、前記描画単位ごとに独立した画像データを生成する描画処理部と、

前記描画単位ごとに生成された前記画像データを統合して最終的に表示すべき画像データを生成する統合部とを含み、

前記分類部、前記描画処理部、および前記統合部は、前記複数の画像処理装置に機能分散されていることを特徴とする画像処理システム。

**【請求項11】**

ネットワークを介して他の装置と情報を交換する画像処理装置であって、

当該装置は、入力された3次元オブジェクトのデータをグループに分類する分類部、同一グループに属する3次元オブジェクトを包含する部分空間をそれぞれ独立した描画単位とする個別の描画処理を行い、前記描画単位ごとに独立した画像データを生成する描画処理部、および前記描画単位ごとに生成された前記画像データを統合して最終的に表示すべき画像データを生成する統合部の内、少なくとも1つの機能構成を含み、

当該装置に含まれない前記機能構成による処理については、前記他の装置からその処理結果を受信して利用することを特徴とする画像処理装置。

**【請求項12】**

重なりを許容して分割された部分空間ごとにその部分空間に含まれる3次元オブジェクトのレンダリングを独立して行い、各部分空間に含まれる3次元オブジェクトのレンダリング結果を奥行き方向の距離を評価して統合することを特徴とする画像処理方法。

**【請求項13】**

1または複数の3次元オブジェクトを1つのグループとして扱い、同一グループに含まれる前記3次元オブジェクトを包含する部分空間の単位で個別にレンダリングを行い、各部分空間のレンダリングの結果を統合して最終的に表示すべき画像データを生成することを特徴とする画像処理方法。

**【請求項14】**

複数の3次元オブジェクトの配列データを読み込む工程と、

表示エリア内に存在する3次元オブジェクトをグループ分けする工程と、

同一グループに属する3次元オブジェクトを包含する部分空間を独立した描画単位として決定する工程と、

前記描画単位ごとに個別のレンダリング処理を行って画像データを生成する工程と、

前記画像データを統合して最終的に前記表示エリア内に表示すべき画像データを生成する工程とをコンピュータに実行させることを特徴とするプログラム。

**【請求項15】**

視点座標系での各3次元オブジェクトの位置を算出し、視点からの距離にもとづいて各3次元オブジェクトの描画に係る詳細度情報を決定する工程をさらに含み、

前記グループ分けする工程は、前記表示エリア内に存在する3次元オブジェクトを前記詳細度情報にもとづいてグループ分けすることを特徴とする請求項14に記載のプログラム。

**【請求項16】**

前記レンダリング処理は、各描画単位の描画処理を複数の描画処理ユニットに分散してなされることを特徴とする請求項14または15に記載のプログラム。

【書類名】明細書

【発明の名称】画像処理方法、画像処理装置、および画像処理システム

【技術分野】

【0001】

この発明は画像処理技術、とくに3次元オブジェクトを2次元表示画面に描画する画像処理方法、画像処理装置、および画像処理システムに関する。

【背景技術】

【0002】

3次元グラフィックスを多用するコンピュータゲームやシミュレーションなどのアプリケーションでは、3次元オブジェクトを見る視点位置が移動したり、ユーザが3次元オブジェクトを移動させたりするため、高速なコンピュータグラフィックス(CG)処理が要求される。具体的には、透視変換によりオブジェクトのワールド座標を2次元座標に変換し、その位置情報とオブジェクトの表面の色や模様などのテクスチャとを用いてレンダリング処理によりピクセル単位のカラー情報を求め、フレームバッファメモリに書き込み、画面に表示するという一連の描画処理をオブジェクトの移動や視点の移動のたびに高速に行う必要がある。

【0003】

レンダリング処理の高速化のために、描画領域を小領域のタイルに分割してタイルごとに並列にレンダリングする分割処理法がある。このタイルレンダリングと呼ばれる方法では、処理の大半をオンチップのメモリで行うことが可能であり、フレームバッファとして用いられる外部メモリへの依存度が低いため、3Dパイプラインを用いた既存のグラフィックチップよりメモリバンド幅が少なくて済む。

【0004】

また、それぞれにVRAMをもった複数のグラフィックプロセッサを並列駆動し、各グラフィックプロセッサによる描画データとZバッファを同期して出力し、外部で合成する方式もある。

【発明の開示】

【発明が解決しようとする課題】

【0005】

タイルレンダリングでは、3Dパイプラインの処理方式と大きく異なるため、アプリケーションとの互換性の問題が生じる他、オブジェクトの数が多くなると、ソート処理のために効率が悪くなるといった問題が生じる。また、グラフィックプロセッサを並列駆動する方式は、並列度は高くなるが、メモリが物理的に分散する構成となるため、VRAMの利用効率が悪く、柔軟な運用が難しい。空間を均一なサイズの標準ビューボリュームに分割して、標準ビューボリューム単位で画像処理を並列化する方法もあるが、3次元オブジェクトの形状や位置関係を無視して空間を分割することになり、オブジェクトごとの性質を反映した柔軟で最適な描画戦略を立てることは難しくなる。

【0006】

このように描画処理の並列度を高めてメモリ効率と演算効率を上げながら、同時に互換性や柔軟性を実現していくことは非常に難しい課題となっている。

【0007】

本発明はこうした課題に鑑みてなされたものであり、その目的は、多様な描画戦略に柔軟に対応することができ、並列処理による効率化が可能な画像処理技術を提供することにある。

【課題を解決するための手段】

【0008】

本発明のある態様は画像処理装置に関する。この装置は、入力された3次元オブジェクトのデータをグループに分類する分類部と、同一グループに属する3次元オブジェクトを包含する部分空間をそれぞれ独立した描画単位とする個別の描画処理を行い、前記描画単位ごとに独立した画像データを生成する描画処理部と、前記描画単位ごとに生成された前

記画像データを統合して最終的に表示すべき画像データを生成する統合部とを含む。

#### 【0009】

ワールド座標系で定義された3次元オブジェクトは、ビューボリュームによりクリッピングされ、平行投影や透視投影などの投影法により2次元の投影面に投影されて表示される。投影面に投影する3次元オブジェクトの数を制限するために、適当な有限領域にビューボリュームを限定することがある。平行投影におけるビューボリュームを投影方向の奥行きによって制限すると、ビューボリュームは平行六面体となる。一方、透視投影におけるビューボリュームを視点位置からの方向、角度、および奥行きによって制限すると、ビューボリュームは四角錐台となる。前記描画処理部は、このように透視投影で規定される四角錐台または平行投影で規定される平行六面体を前記部分空間として利用してもよい。

#### 【0010】

複数の3次元オブジェクトをグループ分けし、同一グループに属する3次元オブジェクトを包含する部分空間に3次元空間を分割した場合、部分空間は重なりをもつことがあるが、このような重なりは許容される。均一なサイズの部分空間に分割すると、3次元オブジェクトを部分空間の境界で分解することになり、境界付近の描画が困難になる。そこで、重なりを許容して、同一グループに属するオブジェクトを包含する不均一なサイズの部分空間に分割し、分割空間ごとに独立した描画戦略を採用することを可能とする。たとえば、分割空間ごとに異なる解像度を選択して、デフォーカス処理を行うといったマルチ解像度描画が可能である。また分割空間ごとに描画処理を独立に行うことができるため、分散描画処理も容易となる。

#### 【0011】

3次元オブジェクトのグループ分けは、オブジェクトの属性にもとづいてなされてもよく、そのオブジェクトを描画する際のLOD (Level Of Detail; 詳細度) に関する情報にもとづいてなされてもよい。LODとは、視距離に応じてオブジェクトをどれくらい詳細に描画するか、そのレベルを与える情報である。オブジェクトの属性やLOD情報はオブジェクトの設計段階であらかじめ設定されていてもよいが、オブジェクトの描画段階においてユーザがマニュアルで設定してもよく、描画システムによって自動的に設定してもよい。

#### 【0012】

前記描画処理部は、複数のレンダリング部を含み、各描画単位の描画処理の複雑度にもとづいて描画処理を前記複数のレンダリング部に分散させてもよい。描画処理の複雑度とは、対象オブジェクトに適用される各種レンダリングアルゴリズムの計算量の多少、そのレンダリングアルゴリズムを実行する上でCPUやメモリなどの計算資源にかかる処理負荷の大小などを含む、広い意味での描画処理の難易性を示すものである。この複数のレンダリング部は、バス結合され、それぞれが専用に利用できるメモリ領域をもち、並列に画像データを生成することのできる並列描画エンジンとして1つのチップ上に実装されてもよい。また、複数のレンダリング部は、当該装置の外部にあって、ネットワークで当該装置に結合している分散処理ユニットの集合体であってもよく、その集合体により各描画単位の描画処理が分散して行われてもよい。

#### 【0013】

この複数のレンダリング部はそれぞれの計算能力やメモリ容量などの処理性能が異なるものであってもよく、各描画単位の描画処理の複雑度に応じた処理性能をもつレンダリング部にその描画処理を行わせてもよい。たとえば、簡単な幾何学処理用のプロセッサと複雑なレンダリング処理用のプロセッサを設けたり、搭載されるメモリの容量を異ならせたりして、レンダリング部の処理性能に違いをもたせ、各描画単位の描画処理の複雑度に応じて、適切なプロセッサやメモリを使い分けてもよい。

#### 【0014】

当該装置の外部にあって、ネットワークで当該装置に結合した分散描画処理ユニットから前記描画単位で描画処理された画像データを受信する通信部をさらに含み、前記統合部は、前記分散描画処理ユニットから受信した画像データを前記描画処理部が生成した画像

データに統合し、最終的に表示すべき画像データを生成してもよい。この分散描画処理ユニットは、当該装置と同じ構成をもち、描画単位ごとに独立した描画処理を行い、画像データを生成する装置であってもよく、これらの装置が互いに通信を行ってそれぞれの描画単位の画像データを交換することで、各装置において最終的に表示すべき画像が生成されるように構成してもよい。

#### 【0015】

各描画単位の描画処理の詳細度に関する情報に応じてネットワーク上の距離の異なる分散描画処理ユニットにその描画処理を行わせてもよい。たとえば、遠景など描画詳細度の要求レベルが低い場合は、当該装置からネットワーク上で距離の離れた分散描画処理ユニットが描画処理を行い、当該装置の通信部が比較的長いレーテンシーで分散描画処理ユニットから描画処理結果を受信してもよい。また、近景など描画詳細度の要求レベルが高い場合は、当該装置からネットワーク上で距離の近い分散描画処理ユニットが描画処理を行い、当該装置の通信部が比較的短いレーテンシーで分散描画処理ユニットから描画処理結果を受信してもよい。

#### 【0016】

本発明の別の態様は、グラフィック処理ブロックと、グラフィック処理ブロックを制御する中央制御ブロックとを单一の素子内に一体化した画像処理装置である。前記グラフィック処理ブロックは、入力された3次元オブジェクトのデータをグループに分類する分類部と、同一グループに属する3次元オブジェクトを包含する部分空間をそれぞれ独立した描画単位とする個別の描画処理を行い、前記描画単位ごとに独立した画像データを生成する描画処理部と、前記描画単位ごとに生成された前記画像データを統合して最終的に表示すべき画像データを生成する統合部とを含み、これらが前記中央制御ブロックからの指示によってグラフィック表示処理を行う。この中央制御ブロックはCPU、メモリ、および割り込み制御やDMA転送を行う制御部を含むCPUコアである。

#### 【0017】

本発明のさらに別の態様は、ネットワークを介して相互に情報を交換しながら、分散描画処理を行う複数の画像処理装置を含む画像処理システムに関する。このシステムは、入力された3次元オブジェクトのデータをグループに分類する分類部と、同一グループに属する3次元オブジェクトを包含する部分空間をそれぞれ独立した描画単位とする個別の描画処理を行い、前記描画単位ごとに独立した画像データを生成する描画処理部と、前記描画単位ごとに生成された前記画像データを統合して最終的に表示すべき画像データを生成する統合部とを含み、前記分類部、前記描画処理部、および前記統合部は、前記複数の画像処理装置に機能分散されている。前記複数の画像処理装置の各々が、前記分類部、前記描画処理部、および前記統合部の機能構成の少なくとも1つを含むように構成して、システム全体で、一連の描画処理が可能なように構成することができる。

#### 【0018】

本発明のさらに別の態様は、ネットワークを介して他の装置と情報を交換する画像処理装置に関する。当該装置は、入力された3次元オブジェクトのデータをグループに分類する分類部、同一グループに属する3次元オブジェクトを包含する部分空間をそれぞれ独立した描画単位とする個別の描画処理を行い、前記描画単位ごとに独立した画像データを生成する描画処理部、および前記描画単位ごとに生成された前記画像データを統合して最終的に表示すべき画像データを生成する統合部の内、少なくとも1つの機能構成を含み、当該装置に含まれない前記機能構成による処理については、前記他の装置からその処理結果を受信して利用する。たとえば、他の装置の分類部による分類結果を受信して、当該装置の描画処理部で利用したり、他の装置の描画処理部による描画処理結果を受信して、当該装置の統合部で利用することができる。また、逆に当該装置の分類部や描画処理部の処理結果を他の装置に送信して、他の装置で利用するようにしてもよい。

#### 【0019】

本発明の別の態様は画像処理方法に関する。この方法は、重なりを許容して分割された部分空間ごとにその部分空間に含まれる3次元オブジェクトのレンダリングを独立して行

い、各部分空間に含まれる3次元オブジェクトのレンダリング結果を奥行き方向の距離を評価して統合する。

【0020】

本発明のさらに別の態様も画像処理方法に関する。この方法は、1または複数の3次元オブジェクトを1つのグループとして扱い、同一グループに含まれる前記3次元オブジェクトを包含する部分空間の単位で個別にレンダリングを行い、各部分空間のレンダリングの結果を統合して最終的に表示すべき画像データを生成する。

【0021】

本発明のさらに別の態様はコンピュータプログラムに関する。このプログラムは、複数の3次元オブジェクトの配列データを読み込む工程と、表示エリア内に存在する3次元オブジェクトをグループ分けする工程と、同一グループに属する3次元オブジェクトを包含する部分空間を独立した描画単位として決定する工程と、前記描画単位ごとに個別のレンダリング処理を行って画像データを生成する工程と、前記画像データを統合して最終的に前記表示エリア内に表示すべき画像データを生成する工程とをコンピュータに実行させる。

【0022】

なお、以上の構成要素の任意の組合せ、本発明の表現を方法、装置、システム、コンピュータプログラム、記録媒体などの間で変換したものもまた、本発明の態様として有効である。

【発明の効果】

【0023】

本発明によれば、3次元オブジェクトの高速で柔軟な描画処理が可能である。

【発明を実施するための最良の形態】

【0024】

本発明の実施の形態を説明するにあたり、まず基礎的な概念を説明する。3次元オブジェクトは3次元ワールド座標で表現されているが、これを画面に描画するためにはワールド座標系の中にビューボリュームを指定し、このビューボリュームによってオブジェクトをクリッピングし、表示画面に投影する必要がある。3次元オブジェクトを2次元上に表示するための投影法の1つに透視投影がある。透視投影によるビューボリュームは一般に視点を頂点とし、視点からの方向と角度によって規定される半無限ピラミッドとなる。本発明では、この半無限のビューボリュームを視点からの方向、角度、および奥行きを制限することによって得られる四角錐台のビューボリュームを利用して、オブジェクトの空間を部分空間に分割する。

【0025】

図1 (a) は、3次元空間における四角錐台のビューボリュームの説明図であり、図1 (b) はその上面図である。視点Oから見た場合の可視領域内で、前方クリッピング面10と後方クリッピング面12により奥行き方向の深さが規定される四角錐台のビューボリューム（以下、ブリックと呼ぶ）が設けられる。投影面14とブリックとの交わり16が表示画面内の描画エリアとなり、このエリア内にオブジェクトが投影されて描画される。以下、奥行き方向をz軸、投影面14と平行な面をx y平面とする。前方クリッピング面10、後方クリッピング面12、投影面14、および背面15は、それぞれ視点Oから距離Z1、Z2、Z3、およびZ4の位置にある。実施の形態では、このようにして規定されるブリックを独立した描画単位として個別の描画処理を行い、ブリックごとに独立した画像データを生成する。さらにブリックごとに生成された画像データを統合して最終的に表示される画像データを生成する。

【0026】

図2 (a) は、3次元空間をブリックにより分割する例を説明する図であり、図2 (b) はその上面図である。第1クリッピング面18と第2クリッピング面20に挟まれた四角錐台の可視領域が4つの互いに重なり合わないブリックで分割されている。第2クリッピング面20と第3クリッピング面22に挟まれた四角錐台の可視領域もまた4つの互い

に重なり合わないブリックで分割されている。第1クリッピング面18、第2クリッピング面20、および第3クリッピング面22は、それぞれ視点Oから距離Z1、Z2、およびZ3の位置にある。このようにx y方向が重なり合わない空間分割を行った場合、タイリングと同様の描画処理が可能であり、小さな高速VRAMエリアを用いて、最終的な大きなサイズの画像を生成することができる。

#### 【0027】

図3 (a)は、3次元空間をブリックにより分割する別の例を説明する図であり、図3 (b)はその上面図である。視点Oから距離Z1の位置にある前方クリッピング面24と距離Z3の位置にある後方クリッピング面26により深さが制限された4つの互いに重なり合わないブリックと、距離Z2の位置にある前方クリッピング面28と距離Z4の位置にある後方クリッピング面30により深さが制限されたブリックとがz方向で互いに重なり合っている。前方のブリックは4分割されているが、後者のブリックは分割されていない。これは一般に手前にあるオブジェクトほど詳細に描画する必要があり、手前のブリックは小さい領域に分割して高い解像度で描画処理するためである。すなわちブリックごとに同サイズのメモリを使って描画処理を行えるように、一般に要求される描画詳細度が低い後方の領域については分割空間を広く取り、描画詳細度が高い前方の領域についてはより小さい空間に分割する。

#### 【0028】

図4 (a)は、さらに別のブリックによる空間分割の例を説明する図であり、図4 (b)はその上面図である。この例では、可視領域内に第1ブリック32、第2ブリック34、第3ブリック36、および第4ブリック38が設けられており、z方向だけでなく、x y方向でも重なりをもっている。第1ブリック32の前方クリッピング面、後方クリッピング面はそれぞれ距離Z1、Z3の位置にあり、第2ブリック34の前方クリッピング面、後方クリッピング面はそれぞれ距離Z2、Z5の位置にある。第3ブリック36の前方クリッピング面、後方クリッピング面はそれぞれ距離Z4、Z7の位置にあり、第4ブリック38の前方クリッピング面、後方クリッピング面はそれぞれ距離Z6、Z8の位置にある。このようにブリックへの分割に重なりがある場合は、ブリックごとの描画処理後、後に述べるZ付マージ描画処理を行って画像データを統合して最終的な表示画像を生成することができる。

#### 【0029】

図5 (a)～(e)は、図4の例の空間分割におけるZ付マージ描画処理の概略を説明する図である。図5 (a)は第4ブリック38における描画処理の結果、図5 (b)は第2ブリック34における描画処理の結果、図5 (c)は第1ブリック32における描画処理の結果、図5 (d)は第3ブリック36における描画処理の結果である。各分割空間における描画データは、適宜拡大または縮小した上で、奥行き値(以下、Z値ともいう)の大小評価にもとづいて1つの画像に統合される。図5 (e)は、図5 (a)～(d)に示すそれぞれのブリックでの描画データをZ付マージ描画処理により統合して得られる画像である。

#### 【0030】

図6は、本発明の実施の形態に係る画像処理装置100の構成を示す。画像処理装置100は、I/Oポート108と、メインCPU110と、メインメモリ112と、通信部114と、グラフィック処理ブロック120とを含み、これらの各ブロックはバス116により接続されている。

#### 【0031】

I/Oポート108は、マウスやキーボードなどの入力装置102、CD-ROMやハードディスクドライブなどの記憶装置104、および画像を表示する表示装置106に対するデータの入出力を制御する。

#### 【0032】

メインCPU110は、画像処理装置100の全体を制御するものであり、一例として128ビットのRISC (Reduced Instruction Set Computer) - CPUである。メイン

CPU110は、記憶装置104に記録されているアプリケーションプログラムを実行する。プログラムの実行により、メインCPU110は、ユーザからの入力に応じてグラフィック処理ブロック120を制御して、画像の表示を制御する。メインCPU110は、とくにグラフィック処理ブロック120に対してマルチパスコマンドによる並列描画制御を行う。

#### 【0033】

メインメモリ112は、記憶装置104から読み出したオブジェクトの形状データやテクスチャデータ等の描画データ、描画処理プログラム、後述のB-box配列やグループリストなどを記憶する。

#### 【0034】

通信部114は、他の画像処理装置100とネットワークを介してデータ通信をする。このようなデータ通信は、他の画像処理装置100と連携して分散処理によって描画処理を実行するために行われる。

#### 【0035】

グラフィック処理ブロック120は、並列描画エンジン122、画像メモリ126、およびこれらを接続するメモリインターフェイス124とを含み、ASIC (Application Specific Integrated Circuit) - DRAM (Dynamic Random Access Memory) として一体に構成されている。

#### 【0036】

並列描画エンジン122は、メインCPU110からのマルチパスコマンドに基づいて、画像メモリ126の全体的な記憶領域を複数の領域に分割し、これらの複数の分割記憶領域を用いて、部分空間ごとの3次元オブジェクトのレンダリングを並列に実行して、それぞれの部分空間における分割画像を生成する。さらに並列描画エンジン122は、並列に描画処理された各分割画像を、いずれか一つの分割領域にZ付マージ描画処理することで、1枚の画像に統合する。レンダリングに用いる隠線、隠面消去の手法として、Zバッファ法、スキャンライン法、レイトレーシング法などが使用される。

#### 【0037】

画像メモリ126は、複数の領域に分割されたDRAMであり、各ピクセルのRGB値および透過度 $\alpha$ を表すA値が格納されるフレームバッファ領域と、各ピクセルの奥行き値を表すZ値が格納されるZバッファ領域、テクスチャマッピングに用いられるテクスチャが格納されるテクスチャ領域に分けられ独立に制御される。また複数のチャネルから独立に描画制御して、複数のブリック単位の描画処理を並列実行できる。一例として、並列描画エンジンは4MBのエリアをもつ8個のDRAMを4チャネルで駆動してもよい。なお、描画エリアとテクスチャエリアは適宜入れ替えることができ、描画した画像をテクスチャとして利用することもできる。

#### 【0038】

たとえば、8つの分割記憶領域のうち4つを描画領域として用い、残りの4つをテクスチャ領域として用いた場合、並列描画エンジン122は4つのテクスチャ領域のそれぞれに格納されたテクスチャデータを処理して、4つの描画領域のそれぞれにZ値が付加された分割画像を描画する。次に、並列描画エンジン122は、各分割画像を、メインCPU110からのマルチパスコマンドにしたがって、いずれかの分割記憶領域にZ付マージ描画処理することで1枚の画像に統合する。

#### 【0039】

図7は、画像処理装置100のグラフィック処理ブロック120の機能構成図である。同図の機能は、主にグラフィック処理ブロック120の並列描画エンジン122、メモリインターフェイス124、および画像メモリ126において実現され、場合によりメインCPU110やメインメモリ112が利用される。同図は機能に着目したブロック図を描いており、これらの機能ブロックはハードウエアのみ、ソフトウエアのみ、またはそれらの組合せによっていろいろな形で実現することができる。

#### 【0040】

オブジェクト記憶部52は3次元オブジェクトデータを格納する。個々のオブジェクトもしくはオブジェクト群を、そのオブジェクトもしくはオブジェクト群の占有する空間領域を包含する球や直方体、多面体などのシンプルな構造体により代表させる。この構造体をB-box (Boundary-box)と呼ぶ。オブジェクト記憶部52はオブジェクトの3次元データの他に、このB-boxの配列データを格納している。分類部44はオブジェクト記憶部52からB-box配列を読み込み、複数のB-boxをオブジェクトの属性やLOD情報にもとづいてグループに分類し、グループリストを作成してオブジェクト記憶部52に格納する。

#### 【0041】

描画処理部46は、透視投影で規定される四角錐台を部分空間として算出して3次元オブジェクトの透視変換を行う座標変換部42と、四角錐台における個別の描画処理を行うレンダリング部43とを含む。座標変換部42は、オブジェクト記憶部52からグループリストを読み込み、同一グループに属するB-boxを包含するブリックを算出し、オブジェクトに対するクリッピング処理、透視変換による座標変換などのジオメトリ処理を行う。レンダリング部43は、ブリックごとに独立した描画処理を行って画像データを生成し、画像記憶部54に格納する。画像データとして画素の色情報データと画素の奥行き値データが生成され、前者はRGBAバッファ、後者はZバッファの形態で格納される。

#### 【0042】

統合部48は、画像記憶部54からブリックごとに生成された画像データを読み込み、画像データの各画素の奥行き値のダイナミックレンジを補正した上で、補正された奥行き値の大小比較により画像データを統合して最終的に表示すべき画像を生成する。生成された画像データは画像記憶部54に格納される。表示部50は画像記憶部54から統合された画像データを読み出し、ディスプレイに表示する。

#### 【0043】

図8は、描画処理部46の並列処理による構成例を説明する図である。描画処理部46は多数オペランドで構成された複数のレンダリングパイプRPで構成される。各レンダリングパイプRPには、オペランド1からオペランド4でそれぞれ特定される、バッファに格納された4つのテクスチャT1からT4のデータが入力される。また、各レンダリングパイプRPは、オペランド5で特定されるRGBAバッファに対するデータD1の読み書きと、オペランド6で特定されるZバッファに対するデータD2の読み書きを行う。各レンダリングパイプRPは、式 $D2 = f2(T1, T2, T3, T4)$ により、後述のようにダイナミックレンジの補正されたZ値を出力する。さらに各レンダリングパイプRPは、式 $D1 = f1(T1, T2, T3, T4, D1, D2)$ により、Z付マージ描画処理を行い、統合された画像データを出力する。

#### 【0044】

図9から図14を参照して、図7の構成のグラフィック処理ブロック120による分割描画処理手順と統合処理手順を説明する。

#### 【0045】

図9は、分割描画処理と統合処理の手順を説明するフローチャートである。3次元オブジェクトのアニメーション処理が行われる(S10)。このアニメーション処理において、運動プログラムにしたがって、視点およびオブジェクトの位置の更新や、オブジェクトの変形が行われる。

#### 【0046】

分類部44は、アニメーション処理されたオブジェクトについてのB-boxの配列をオブジェクト記憶部52から読み込み(S12)、視点座標系における各B-boxの位置を算出し、クリップ情報と視点からの距離からLOD情報を決定する(S14)。LOD情報はオブジェクトをどれくらい詳細に描画するかを示すレベルであり、一般に視点からの距離によってその描画レベルが決定される。たとえば視点からの距離が遠いオブジェクトについては低いLODに決定し、視点からの距離が近いオブジェクトについては高いLODに決定する。

## 【0047】

さらに分類部44は、クリップ情報により描画エリア内にあると判断されたB-b ox群のオブジェクト属性とLOD情報を調べ、オブジェクト属性とLOD情報が同一または類似するものをグルーピングし、グループリストを作成する(S16)。

## 【0048】

ここで、オブジェクト属性には、そのオブジェクトの運動が静的であるか、動的であるかといった運動特性を示す情報の他、背景、人物、建物などオブジェクトの種類を示す情報などが含まれる。このオブジェクト属性は、オブジェクトの設計者がオブジェクトのグループ分けを想定してあらかじめ設定したグループ識別情報であってもよい。たとえば設計者が同一グループに分類すべきオブジェクトについては同一のグループ識別情報を設定し、異なるグループに分類すべきオブジェクトについては、異なるグループ識別情報を設定しておくことができる。

## 【0049】

図10(a)はB-b ox配列のグループ分けの例である。B-b ox配列の各要素の内、描画エリア内にあるものについて、グループ分けされている。たとえばB-b ox配列の要素a1、a2、およびa3はグループAに分類されており、要素b1およびb2はグループBに分類されている。図10(b)は、図10(a)のグループ分けにより生成されるグループリストである。グループAの要素a1～a3、グループBの要素b1およびb2、グループCの要素c1～c5、グループDの要素d1およびd2が順に並んだリストとなっている。

## 【0050】

再び図9を参照し、描画処理部46は、グループリストにもとづいて同一グループに属するB-b ox群を包含するブリックを算出し、グループリスト内の対応するグループに関連づける(S18)。

## 【0051】

図11(a)は、図2の空間分割の例を用いて、同一グループのB-b ox群を包含するブリックを説明する図であり、図11(b)はその上面図である。第1クリッピング面18と第2クリッピング面20に挟まれた領域にある3次元オブジェクト60を包含するB-b ox62は1つのグループに属し、このB-b ox62を包含する部分空間として前方クリッピング面18と後方クリッピング面20で挟まれたブリック63が算出される。同様に2つのB-b ox64および66は別のグループに属し、これらを包含する部分空間として第1クリッピング面18と第2クリッピング面20で挟まれたブリック67が算出される。

## 【0052】

一方、第2クリッピング面20と第3クリッピング面22に挟まれた領域では、1つのB-b ox68を包含する部分空間としてブリック69が算出され、他の2つのB-b ox70および72を包含する部分空間として別のブリック73が算出される。なお説明を簡単にするため、他のブリックに含まれるB-b oxは図示していない。

## 【0053】

このように、視点から距離Z1からZ2の位置にある3つのB-b ox62、64および66と、視点から距離Z2からZ3の位置にある3つのB-b ox68、70および72は、主にLOD情報の違いから、z方向に重なりのない別々の部分空間に属することになる。さらにより小さい領域において分割描画処理を行うことができるよう、x y方向でも重なりのない部分空間に4分割される。このように3次元オブジェクトの占有する空間をx、yおよびz方向で重なりのない部分空間に分割することができる場合、オブジェクトがZ境界に存在しない限り、Z付マージ描画処理とz方向のシザリング処理を省略して各ブリックの描画処理を独立して行うことができる。もっとも、オブジェクトが大きすぎるなどの理由により、オブジェクトを分割して複数のブリックに属させる必要が生じることもあり、その場合はz方向のシザリング処理を要する。

## 【0054】

図12 (a) は、図3の空間分割の例を用いて、同一グループのB-b ox群を包含するブリックを説明する図であり、図12 (b) はその上面図である。視点Oから距離Z1の位置にある前方クリッピング面24と距離Z3の位置にある後方クリッピング面26に挟まれた領域にある2つのB-b ox74および76は1つのグループに属し、これらを包含する部分空間として、前方クリッピング面24と後方クリッピング面26に挟まれたブリック77が算出される。同様に他の2つのB-b ox78および80は別のグループに属し、これらを包含する部分空間として、前方クリッピング面24と後方クリッピング面26に挟まれたもう1つのブリック81が算出される。

#### 【0055】

一方、視点Oから距離Z2の位置にある前方クリッピング面28と距離Z4の位置にある後方クリッピング面30に挟まれた領域には1つのB-b ox82があり、これを包含する部分空間として前方クリッピング面28と後方クリッピング面30に挟まれたブリック83が算出される。

#### 【0056】

この例では、2つのB-b ox80とB-b ox82がz方向において重なりをもち、それらが異なるグループに属するため、前方のブリック77および81と、後方のブリック83との間にz方向の重なりが生じる。このようにz方向の重なりを許容した部分空間への分割により、オブジェクトがZ境界に存在するような場合でも、z方向にオブジェクトを分解することなく、そのオブジェクトをブリック単位の描画処理により表現することができる。さらに、凸オブジェクトであれば、Z付マージ描画処理を省略することができる。また、この例では、x y方向には重なりをもたないので、シザリング処理によりブリックごとに分割して描画処理することが可能である。

#### 【0057】

図13 (a) は、図4の空間分割の例を用いて、同一グループのB-b ox群を包含するブリックを説明する図であり、図13 (b) はその上面図である。視点Oに最も近い位置にあるB-b ox84を1つのグループとして、これを包含する部分空間としてブリック32が算出され、以下同様に2つのB-b ox86および88を1つのグループとして、これらを包含する部分空間としてブリック34が、さらに他の2つのB-b ox90および92を1つのグループとして、これらを包含する部分空間としてブリック36が算出され、最後に視点Oから最も離れた位置にあるB-b ox94を1つのグループとして、これを包含する部分空間としてブリック38が算出される。

#### 【0058】

このようにして算出された4つのブリック32、34、36、および38はz方向だけでなく、x y方向においても重なりをもっている。x y方向の重なりを許容した部分空間への分割であるため、x y方向においてもオブジェクトを単位とした空間分割が可能となり、シザリング処理による負荷を軽減することができる。

#### 【0059】

図9に戻り、描画処理部46は、各ブリックに含まれるオブジェクトの属性とLOD情報にもとづいて、各ブリックの描画エリアと描画処理を選定し、ブリック間の依存関係から並列処理への分散とスケジューリングを決定し、処理手順リストを作成する (S20)。

#### 【0060】

図14は、処理手順リストの例であり、レンダリングパイプRP1～RP4の各々に割り振られるグループA～Gが列挙されている。描画処理部46は、ブリックごとに個別のレンダリング処理を行うことができる。ブリック単位で決定されたLODに違いがある場合、ブリックによってレンダリング処理の複雑さが異なる。またブリックに属するオブジェクトの特性や種類によって異なるレンダリングアルゴリズムを利用することもある。そこで複雑で時間のかかるレンダリング処理が必要なブリックと、単純なレンダリング処理で対処できるブリックとに分けて処理の分散を図るなど、ブリックごとの処理の複雑さに依存して処理の分散の仕方を決定することができる。また、半透明のオブジェクトが存在

する場合、重ね合わせる際に順序関係が生じる。その場合、重ね合わせの順序に合わせて個別の描画処理が進むようにスケジューリングを決定する。

#### 【0061】

描画処理部46は、処理手順リストに従い、ブリック単位の描画処理を行う（S22）。描画処理部46は、ブリック単位で個別の描画戦略を選択することができる。たとえば、ブリックごとに異なる解像度を選択してマルチ解像度描画を行うことができる。デフォーカス処理を行う場合、フォーカス処理に応じた空間分割を行い、ブリックごとにデフォーカス表示に応じた情報量の解像度を選択することにより、適切なメモリ量で高速なデフォーカス処理が可能となる。

#### 【0062】

また、描画処理部46は、ブリック単位で個別の描画アルゴリズムを適用することもできる。たとえば、レンダリングに用いる隠線、隠面消去の手法として、Zバッファ法、スキヤンライン法、レイトレーシング法などを適宜選択して適用することができる。また、シェーディングの手法として、フラットシェーディング法、グローシェーディング法、スムーズシェーディング法などを選択的に適用することができる。

#### 【0063】

再び図9を参照し、統合部48は、ブリックごとの描画結果をZ付マージ描画処理により統合して最終的な出力画像を生成する（S24）。その後、ステップS10に戻り、次のアニメーション処理を行い、分割描画処理から統合処理までの一連のレンダリング処理を繰り返す。

#### 【0064】

分割描画による画像データを最終的な出力画像に統合する際、図5で既に説明したように、個々のブリックの画像データに対して、拡大、縮小などのアフィン変換処理を施す。その際、分割描画の画像データの解像度を統合後の画像データの解像度に合わせるために、各ブリックの画像データの画素値を適宜補間する必要があり、内分を用いて画素値を近似する双1次（bilinear）補間や、3次スプライン関数などで画素の変化を近似する双3次（bicubic）補間などが用いられる。

#### 【0065】

図15（a）は、双1次補間による画像データの生成処理を説明する図である。最終的に表示する出力画像として点Tの位置に画素データを生成する必要があるが、ブリック単位の描画データにはその位置に対応する画素データがなく、その近傍に4個の画素データP、Q、RおよびSがあるとする。その場合、4個の画素データP、Q、RおよびSの画素値を補間して、最終画像の点Tの位置の画素データを生成する。説明上、これらの画素データを、その座標位置がx y 平面に格子状に設けられ、その画素値がz軸方向の高さである3次元空間における点として扱う。平面y=0上で2点P、Qを内分して点Aが得られ、平面y=1上で2点S、Rを内分して点Bが得られる。次に点Tを通り、x軸に垂直な平面上で内分点A、Bをさらに内分して点Tの高さすなわち画素値が得られる。

#### 【0066】

図15（b）は、双3次補間による画像データの生成処理を説明する図である。最終的に表示する出力画像として点Tの位置に画素データを生成する際、その周囲にある16個の画素データを用いた補間ににより画素値を算出する。平面y=0上で4点P、Q、RおよびSを通る曲線を3次スプライン関数により求め、同様に、平面y=1、y=2、およびy=3上で、x軸方向に並ぶ4点を通る曲線をそれぞれ求める。次に点Tを通り、x軸に垂直な平面とこの4本の曲線との4つの交点を求める。さらに、この4つの交点を通る曲線を求め、点Tを通りy軸に垂直な平面とこの曲線との交点のz座標により点Tの高さすなわち画素値を得る。

#### 【0067】

図16（a）～（c）はZ付マージ描画処理を説明する図である。図16（a）のように重なりのある4つのブリック170a、172a、174a、および176aでオブジェクトの空間が分割されたとする。図16（b）は、図16（a）の4つのブリック17

0 a、172 a、174 a、および176 aのそれに対応する描画データのZ値の分布170 b、172 b、174 b、および176 bを示す。このZ値は各ブリック内で局所的に規定される深さであり、分割前の元の空間での絶対的な深さを示すものではない。各ブリックの描画データを統合するには、Z値のダイナミックレンジを元の空間における絶対的な値に補正する必要がある。

#### 【0068】

図16 (c) は、各ブリックのダイナミックレンジの補正されたZ値と、それらを統合して得られる元の空間における絶対的なZ値の分布を示す。統合後の描画データのZ値の範囲178に対して、4つのブリック170 a、172 a、174 a、および176 aのそれに対応する補正されたZ値の範囲170 c、172 c、174 c、および176 cが示されており、これらの補正されたZ値の範囲に適合するように各ブリック内でローカルに規定されたZ値のダイナミックレンジが補正される。図16 (b) の4つの描画データのZ値の分布170 b、172 b、174 b、および176 bのそれに対応する補正されたZ値の分布170 d、172 d、174 d、および176 dが示されている。これらの補正されたZ値の分布170 d、172 d、174 d、および176 dを統合したものが、最終的な描画データのZ値の分布180である。

#### 【0069】

このようにZ値のダイナミックレンジが補正された後、Zバッファ法により隠面消去されて最終的な統合画像が描画される。Zバッファ法により画素単位の描画を行う際、各画素のA値を用いてアルファブレンディングを行い、半透明の合成画像を生成することもできる。

#### 【0070】

ブリック内でローカルに規定された深さzは、一次変換式 $Z = (a/A) \times z + b$ により、分割前の元の空間の深さZに変換される。ここで定数Aはブリック内でローカルに規定される深さの最大値であり、定数aは分割前の元の空間での当該ブリックの深さの最大値である。定数bは視点から当該ブリックのクリップ面までの距離である。

#### 【0071】

以上説明したように、実施の形態に係る画像処理装置は、3次元オブジェクトを属性やLODにもとづいてグループ分けし、同一グループのオブジェクトを含むブリックに空間を分割して、ブリックごとに個別のレンダリング処理を行うことができる。空間分割により処理を並列化して描画処理を高速にすると同時に、オブジェクトの特性に合わせたレンダリング処理を適用することができる。またブリックの大きさをある程度の大きさに制限することで、ブリック単位の描画処理に要するVRAMの大きさを一定サイズにすることができる。また3次元オブジェクトをグループ分けすることにより、テクスチャやオブジェクトモデルのデータをグループ単位で一括管理することができ、レンダリングアルゴリズムの互換性やバージョンの管理が容易になる。

#### 【0072】

以上、本発明を実施の形態をもとに説明した。これらの実施の形態は例示であり、それらの各構成要素や各処理プロセスの組合せにいろいろな変形例が可能のこと、またそうした変形例も本発明の範囲にあることは当業者に理解されるところである。

#### 【0073】

そのような変形例を説明する。実施の形態では透視投影で規定される四角錐台で空間を分割したが、空間分割はこれに限らず、平行投影で規定される平行六面体で分割してもよく、本発明の分割描画処理は、特定の投影法に限定されない。

#### 【0074】

分割空間単位のオブジェクトのレンダリング処理はデータの局所性が保証され、分割空間の相互の依存関係については、個々の分割空間の描画データをZ付マージ描画処理により統合することで対処することができる。そこで、分割空間単位の描画処理をネットワークで結合された計算ノードに分散して行い、各ノードで計算された描画データを中央の計算ノードに集めて統合し、最終的な出力画像を生成してもよい。この計算機ネットワーク

は、計算ノードがブロードバンドのネットワークにより、ピア・ツー・ピアで接続されたものであってもよい。またそれぞれの計算ノードがセルとして全体のネットワークシステムに貢献し、それぞれのセルのオペレーティングシステムが互いに連携することで、全体として一つの巨大なコンピュータとして動作するように構成してもよい。本発明の分割描画処理は、このような計算機ネットワークを前提とした分散処理にも適しており、これにより大規模な計算資源を必要とする画像処理が可能である。

#### 【0075】

さらに、このようなネットワークを前提とした分散描画処理システムでは、中央の計算ノードと分散された計算ノードの間で、ルーティングのホップ数や通信のレーテンシーなどにもとづいてネットワーク上の距離を定義することができる。そこで、分割空間単位で描画されるオブジェクトの視点からの距離と、中央の計算ノードから分散された計算ノードまでの距離とを対応づけ、分割空間単位の描画処理を計算ノードに分散させることもできる。すなわち、視点から近い位置にあるオブジェクトは、視点の移動やオブジェクト自身の移動により頻繁に画像データが更新されるため、ネットワーク上でも近い位置にある計算ノードで描画処理を行い、描画結果を短いレーテンシーで中央ノードに集める。一方、視点から遠い位置にあるオブジェクトは、視点の移動やオブジェクト自身の移動によってあまり変化しないため、ネットワーク上で遠い位置にある計算ノードで描画処理してもよく、描画結果が中央ノードに届くまでのレーテンシーが長くても問題とならない。

#### 【0076】

実施の形態では、オブジェクトのグループ分けにもとづいて算出されたブリック間にz方向の重なりがある場合、Z付マージ描画処理を行ったが、Z付マージ描画処理が不要となるように、言い換えればブリック間のz方向の重なりが生じないように、オブジェクトのグループ分けを決定してもよい。以下、図17から図20を参照してこの変形例を説明する。

#### 【0077】

図17 (a) は、同一グループに属する2つのB-box 200 および202とそれらを包含する部分空間として算出されたブリック204を説明する図であり、図17 (b) はその上面図である。2つのB-box 200 および202は、たとえば3次元空間を高速に移動するオブジェクト（以下、高速オブジェクトという）をその中に含み、この運動特性にもとづいて同一グループに分類されたものである。上記の実施の形態では、これら2つのB-box 200 および202を包含する部分空間として、同図のように視点Oから距離Z1の位置にある前方クリッピング面と距離Z2の位置にある後方クリッピング面に挟まれたブリック204が算出される。

#### 【0078】

図17 (a)、(b) のようにブリック204が定められ、描画処理が進められた後、図18 (a)、(b) に示すように、別の運動特性をもつオブジェクトを含む新たなB-box 206がブリック204の中に進入してきたとする。図18 (a) はこの新しい状況下での空間分割を説明する図であり、図18 (b) はその上面図である。新たなB-box 206は、たとえば3次元空間を低速に移動するオブジェクト（以下、低速オブジェクトという）をその中に含むものであり、その運動特性の違いから高速オブジェクトを含む2つのB-box 200 および202とは別のグループに分類される。したがって上記の実施の形態では、低速オブジェクトを含む新たなB-box 206を包含する部分空間として、同図のように視点Oから距離Z2の位置にある前方クリッピング面と距離Z3の位置にある後方クリッピング面に挟まれたブリック208が新たに算出され、高速オブジェクトを含む2つのB-box 200 および202を包含するブリック204とz方向の重なりをもつようになる。

#### 【0079】

このようにブリック間でz方向の重なりがあっても、上記の実施の形態によれば分割空間ごとに描画処理を行い、Z付マージ描画処理によって統合することができる。しかし、計算性能の制約などからZ付マージ描画処理を行えないような状況や、Z付マージ描画処

理を省いて描画処理を簡略化したい場合には、Z付マージ描画処理が不要となるように、B-b0x群のグループ分けを工夫し、z方向の重なりのないブリックを算出してもよい。図19は、z方向の重なりを解消することのできるブリックの算出手順を示すフローチャートである。このフローチャートは、図9の分割描画処理と統合処理の手順を示すフローチャートのステップS18の変形であり、ステップS16までと、ステップS20以降は図9と同じである。

#### 【0080】

ステップS16において、分類部44によりB-b0x群がグループ分けされた後、描画処理部46により同一グループに属するB-b0x群を包含するブリックが算出される(S30)。描画処理部46は、算出されたブリックがz方向に重なりをもつかどうかを調べる(S32)。z方向に重なりがある場合(S32のY)、分類部44はz方向に重なりがあるブリックについてその重なりを解消するようにB-b0x群を再グループ分けする(S34)。

#### 【0081】

図20(a)はB-b0x群の再グループ分けを説明する図であり、図20(b)はその上面図である。高速オブジェクトを含む2つのB-b0x200および202はこれまで同一グループに分類されていたが、このままでは図18(a)、(b)で説明したようにブリック間にz方向の重なりが生じる。そこで分類部44は、この重なりを解消する目的で、2つのB-b0x200および202を別々のグループに分類する。その結果、ステップS30に戻って、描画処理部46が同一グループに属するB-b0x群を包含するブリックを再度算出すると、図20(a)、(b)に示すように、別々のグループに属するB-b0x200および202のそれぞれを包含する2つのブリック210および212が算出される。このため、新たに加わったB-b0x206を包含するブリック208とはz方向の重なりがなくなる。図19に戻り、z方向のブリックの重なりが解消された場合(S32のN)、ステップS20に進み、それ以降の分割描画処理と統合処理がなされる。

#### 【0082】

このように再グループ分けによりz方向に重なりのない空間分割がなされ、Z付マージ描画処理を不要とすることができます。この再グループ分けの必要性の判断は、図9のステップS20のブリック単位の描画処理の選定時になされてもよい。すなわち、分類部44によりブリックに分割された後に、描画処理部46がブリックごとの描画処理の複雑度を判断し、その複雑度にもとづいて分類部44に再グループ分けを指示してもよい。たとえば予想される計算コストが所定値より大きい場合に再グループ分けを指示する。

#### 【0083】

さらに別の変形例として、実施の形態に係る画像装置100をネットワークに接続されたパーソナルコンピュータ(PC)などの計算ノードとして構成し、複数の計算ノードにおいてオブジェクトの描画処理を行い、ネットワーク経由で画像データをやりとりして画像データを統合する分散描画システムを構成することもできる。

#### 【0084】

図21は、分散描画処理システムの構成例を説明する図である。3台のPC1～PC3とサーバ7がネットワーク6に接続している。サーバ7は、ユーザがログインして仮想キャラクタを操作することのできる仮想空間のサービスを提供する。PC1～PC3は、それぞれ上述の実施の形態に係る画像処理装置100に相当し、サーバ7と通信して、サーバ7の提供する仮想空間のサービスを受けて、仮想空間の描画処理を行うとともに、他のPCと通信して、分散描画処理を行う。

#### 【0085】

PC1では、ユーザAが仮想空間にログインして、仮想空間内で仮想キャラクタC1を操作し、ユーザAの視点から見た仮想空間の画像が生成される。PC2では、ユーザBが同じ仮想空間にログインして、仮想キャラクタC2を操作し、ユーザBの視点から見た仮想空間の画像が生成される。PC3には、画像情報とともに視点からの距離情報を取得可

能なカメラ5が接続されており、カメラ5の視点から広視野角で撮影した遠景の画像情報と距離情報がPC3に入力され、PC3において描画処理される。

#### 【0086】

PC1～PC3のオブジェクト記憶部52には、それぞれの視点を中心にLOD情報にもとづいてオブジェクト群を階層化したB-boxのマップが格納される。B-boxマップは、階層が高いほど遠くから見た場合のオブジェクト群の情報をもち、粗い詳細度でオブジェクト群の描画がなされ、階層が低いほど、近くから見た場合のオブジェクト群の情報をもち、より細かい詳細度でオブジェクト群の描画がなされる。B-boxマップに含まれるオブジェクト群のデータは個々のPCやサーバで分散管理されており、B-boxマップには、個々のオブジェクトの管理元であるPCまたはサーバのアドレスが含まれる。このB-boxマップはユーザの視点の移動とともに更新される。たとえば、ユーザの視点が遠ざかれば遠景のB-boxマップとなり、視点が近づけば近景のB-boxマップとなる。PC1～PC3はそれぞれがもつB-boxマップにもとづいてオブジェクト群の描画処理を行い、必要に応じて生成された画像データを交換する。

#### 【0087】

PC1は、ユーザAが仮想空間を観察する際の視点に関する情報をPC2およびPC3にネットワーク6を介して送信する。PC2およびPC3の描画処理部46はそれぞれ、ユーザAの視点から観察された場合の仮想キャラクタC2および遠景の描画処理を行って画像データを生成し、PC1に送信する。PC1の統合部48はPC2およびPC3から受信した仮想キャラクタC2および遠景の画像データと、PC1で描画処理された仮想キャラクタC1の画像データとをマージして、画像記憶部54に格納する。PC1の表示部50は画像記憶部54からマージされた画像データを読み出し、ディスプレイ4に表示する。

#### 【0088】

ここで、ユーザAの仮想キャラクタC1とユーザBの仮想キャラクタC2が互いに接近している場合は、オブジェクトの重なりや相互作用などの干渉が起こるため、PC1とPC2の間では視点情報以外に、3次元オブジェクトデータを交換して干渉部位の描画処理を行う。

#### 【0089】

また、ユーザBの仮想キャラクタC2がユーザAの視野内に常に存在するとは限らないため、仮想キャラクタC2がユーザAの視野内に入ったときだけ、PC2は仮想キャラクタC2の画像データをPC1に送信する。PC1が仮想キャラクタC2の位置情報を把握している場合は、仮想キャラクタC2がユーザAに視野内に入ったときのみ、ユーザAの視点情報をPC2に送信し、PC2は、ユーザAの視点から見た場合の仮想キャラクタC2の描画データを生成し、PC1に送信する。

#### 【0090】

PC1が仮想キャラクタC2の位置情報を把握していない場合は、PC1は常にユーザAの視点情報をPC2に送信し、PC2は、その視点情報にもとづいてユーザAの視野内に仮想キャラクタC2が存在するか否かを判定し、視野内に存在する場合のみ、ユーザAの視点から見た場合の仮想キャラクタC2の描画データを生成し、PC1に送信する。仮想キャラクタC1と仮想キャラクタC2が相互に干渉する機会が少ない場合、PC1とPC2間での視点情報や3次元オブジェクトデータのやりとりなどの相互通信の時間間隔を大きくとることができる。この干渉性の判断は、オブジェクトを含むB-boxの大きさとB-box間の距離にもとづいてなされる。なお、サーバまたは第三者のPCがPC1におけるユーザAの視点情報と仮想キャラクタC2の位置情報を監視するようにしてもよい。

#### 【0091】

図22は、図21のPC1のディスプレイ4に表示される画像の例を示す。遠景には山234のような静止オブジェクトと、雲232や飛行機230のような運動オブジェクトが表示されている。この遠景は、PC3のカメラ5により撮影され、PC1のユーザAの

視点情報にもとづいて描画処理された画像データである。近景にはPC1で操作されている仮想キャラクタC1と、PC2で操作されている仮想キャラクタC2が表示されている。仮想キャラクタC1、C2はそれぞれPC1、PC2において描画処理された画像データである。

#### 【0092】

図23(a)は、PC1の描画処理部46が算出するブリックを説明する図であり、図23(b)はその上面図である。PC1の描画処理部46は、図22の2つの仮想キャラクタC1およびC2をLOD情報とオブジェクトの属性にもとづいて同一グループに分類し、さらに、2つの仮想キャラクタC1およびC2が接近していることから、干渉性を考慮して、これらのオブジェクトのB-box240および242を包含する部分空間を近景の一つの描画単位として算出する。すなわち、同図のように視点Oから距離Z1の位置にある前方クリッピング面と距離Z2の位置にある後方クリッピング面に挟まれたブリック220が算出される。

#### 【0093】

PC1の統合部48は、リアルタイムでPC2から仮想キャラクタC2のオブジェクトデータを受信して、ブリック220内で仮想キャラクタC1と仮想キャラクタC2の干渉部位を考慮した描画処理を行い、ブリック220の最終的な画像データを生成する。

#### 【0094】

PC1の描画処理部46は、さらに、遠景の描画単位となるブリックを算出する。図22の山234、雲232、および飛行機230はともに遠景に合成されるオブジェクトであり、LODの観点からは共通するが、静止オブジェクトの山234と低速移動オブジェクトの雲232は、高速移動オブジェクトの飛行機230とは運動特性が異なる。そのため、描画処理部46は、飛行機230を山234および雲232とは異なるグループに分類し、遠景のオブジェクトを2つのグループに分けて、それぞれのグループのオブジェクトのB-boxを包含する部分空間を遠景の描画単位として算出する。すなわち、同図のように視点Oから距離Z3の位置にある前方クリッピング面と距離Z4の位置にある後方クリッピング面に挟まれたブリック222と、ブリック222内に含まれる別の描画単位であるブリック224とが算出される。前者のブリック222は、山234および雲232のB-boxを包含する部分空間であり、後者のブリック224は、飛行機230のB-boxを包含する部分空間である。

#### 【0095】

PC1の統合部48は、PC3の描画処理部46が生成する遠景のブリック222および224の画像データを受信して、近景のブリック220の画像データと統合して最終的に表示される画像を生成する。PC3から取得する遠景の画像データは必ずしも3次元グラフィックスデータでなくてもよく、JPEG (Joint Photographic Experts Group)などの2次元グラフィックスデータであってもよい。その場合は、PC1の統合部48は、単に2次元の背景画像に近景のブリック220の画像データをはめ込む処理を行う。

#### 【0096】

ここで、PC3の描画処理部46は、高速移動オブジェクトを含む遠景のブリック224の画像データをオブジェクトの移動速度に合わせて頻繁に更新し、PC1に送信しなければならないのに対して、静止オブジェクトや低速移動オブジェクトを含む遠景のブリック222の画像データについては、それほど頻繁に更新してPC1に送信する必要はない。また、高速移動オブジェクトの通過後は、遠景は、静止オブジェクトと低速移動オブジェクトを含むブリック222のみになるため、PC3からPC1へのデータの送信時間間隔は長く取ることができる。したがって、通常は、遠景を描画する計算ノードと近景を描画する計算ノードの間では、通信のレーテンシーが長くても、分散描画処理システムは全体として機能する。

#### 【0097】

なお、2つの仮想キャラクタC1およびC2の間に干渉性がない場合は、PC1の描画処理部46は、仮想キャラクタC1およびC2をそれぞれ個別に包含する部分空間を描画

単位として算出する。すなわち、図24（a）、（b）に示すように、仮想キャラクタC1のB-box240を包含するブリック226と、仮想キャラクタC2のB-box242を包含するブリック228が算出される。

#### 【0098】

この場合、PC1の統合部48は、リアルタイムでPC2の描画処理部46が生成する仮想キャラクタC2を含むブリック228の画像データを受信して、仮想キャラクタC1を含むブリック226の画像データと統合して、近景の画像データを生成する。

#### 【0099】

このように、PC1のユーザAから見た場合の仮想空間を近景と遠景のブリックに分けたり、オブジェクトの運動特性などの性質にもとづいてブリックを分けることにより、複数の計算ノードでそれぞれのブリックのLODの要求仕様に応じた描画処理を実行したり、計算ノード間の通信頻度を変えるなど、ブリックごとに柔軟に描画戦略を立てて分散描画処理を行うことができる。

#### 【0100】

本発明では、ブリックごとに独立に描画戦略を選択することが可能であり、オブジェクトの特性やLODの要求仕様などに応じてブリック単位で描画アルゴリズムを限定して、描画アルゴリズムの切り替えコストを削減することができる。言い換れば、一様な描画戦略を適用することができる部分空間をブリックとして抽出することにより、ブリック内で必要なプログラムやデータの種類を限定して、描画処理を高速化し、描画アルゴリズムに適したサイズのメモリ領域を割り当てて、メモリの利用効率を高めることができる。すなわち、ブリックを用いることにより、描画戦略の自由度が高まると同時に、演算効率とメモリ効率を上げることができるのであり、さらにいくつかの具体例を挙げてそのことを説明する。

#### 【0101】

図25（a）～（c）は、移動オブジェクト348にモーションブラー処理を施す場合のメモリ領域の選択を説明する図である。同図において、移動オブジェクト348の移動方向は矢印で示されている。通常であれば、図25（a）のように描画エリア340が割り当てられるところ、描画処理部46は、移動オブジェクト348を含むブリックの描画処理にあたって、図25（b）のように移動オブジェクト348の速度方向に圧縮した描画エリア342をメモリに割り当て、速度方向に解像度を減らして描画処理を行う。最終的な出力画像に統合する際、統合部48が、図25（c）のように速度方向に描画エリア346を伸長する。移動オブジェクト348の速度方向には高い解像度は必要ではないため、移動オブジェクト348を含むブリックに対して、速度方向に圧縮した描画エリア342を選択することにより、そのブリックの描画処理の計算量とメモリ量を低減して、描画処理の高速化を図ることができる。

#### 【0102】

図26（a）、（b）は、デフォーカス処理を行う場合のメモリ領域の選択を説明する図である。描画処理部46は、図26（a）のように、デフォーカスすべきオブジェクトを含むブリックについては、低解像度の描画エリア360を割り当て、描画処理を行い、オブジェクトの描画データ364を生成する。最終的な出力画像に統合する際、統合部48は、図26（b）のように、描画エリア362を拡大し、オブジェクトの描画データ366を補間により生成する。たとえば、点368の位置のピクセル値をその周囲にある描画エリア362の格子点のピクセル値を用いて双3次補間などにより補間して求める。このように、デフォーカス処理を施すオブジェクトの描画には高い解像度を要しないため、デフォーカスすべきオブジェクトを含むブリックに対しては、低解像度の描画エリア360を選択する。ブリックごとのフォーカスの深度に応じて、描画エリアの解像度を適正なレベルに調整して描画処理を行うため、必要以上の演算やメモリへのアクセスを行うことを避けて、描画処理を高速化することができる。

#### 【0103】

例示したように、ブリックごとにオブジェクトの運動速度などの特性や、LODの要求

仕様などに応じて解像度を適切に選択してマルチ解像度描画を行うことで、演算量とメモリ量の両面で描画処理を最適化することができる。

#### 【0104】

ブリック単位の描画戦略の他の例として、あるブリックが他のブリックとz方向に重なりをもたず、ただ一つの凸オブジェクトを含むものである場合は、そのブリックについてはZ付マージ描画処理が不要であり、描画処理部46は、Zバッファを使用しないでそのブリックの描画処理を行うことができる。これにより、Z値参照によるレーテンシーをなくして描画処理を高速化できる。描画空間全体に対して均質な解像度のZバッファを用いるのではなく、ブリックごとにZバッファの解像度を変更したり、この例のようにZバッファそのものを使用しないで描画することもマルチ解像度描画の一例ととらえることができる。

#### 【0105】

また、ブリック単位の描画処理はZカリング（Z-culling）処理に有効に適用することができ、Zカリング処理の戦略をブリック単位で調整してZカリング処理の自由度を高め、かつ、処理効率を向上することができる。以下、ブリック単位のZカリング処理について、図27～図33を参照して説明する。

#### 【0106】

Zバッファによる隠面消去法では、ピクセル単位でオブジェクトの奥行きを比較し、どのオブジェクトが前面に位置するかを判定する。一方、Zカリング処理では、一般に階層化されたZバッファを利用して、いくつかのピクセルの集合であるブロックの単位でZ値の代表値をもたせ、ブロック単位のZ値の比較により、オブジェクトの前後関係を判定する。Zカリング処理では、Zバッファによるピクセルレベルでの隠面消去に先立ち、オブジェクトの外形をブロック単位の矩形領域で近似して、ブロックレベルでオブジェクトの奥行き情報を把握し、前方のオブジェクトの背後に隠れる後方のオブジェクトを早い段階で検出して、その後方のオブジェクトのレンダリングを省略する。

#### 【0107】

図27はZカリング処理で用いられる階層化Zバッファの説明図である。同図では3階層のZバッファを模式的に表したものであり、最下位層のZバッファ250の4ピクセル256a～dは一つ上の階層のZバッファ252においては一つのピクセル258により代表される。この代表ピクセル258のZ値は、下位層の4ピクセル256a～dの値の最大値に設定される。さらに、この階層のZバッファ252における4つのピクセルは、同様にもう一つ上の階層のZバッファにおいて一つのピクセル259により代表される。

#### 【0108】

このようにして、階層化Zバッファでは、階層を上がるごとに描画領域の分割単位が粗くなり、各分割単位のZ値は、一つ下の階層における対応領域の一定数のピクセル、ここでは4ピクセルのZ値の最大値で与えられる。したがって、この例では、最上位層の1つのZ値は、最下位層の16ピクセル分のZ値を代表するものとなり、データ量が16分の1に減少する。一般に、Zバッファ法では、スクリーンと同じ解像度のZバッファが必要であり、大容量のメモリを必要とし、メモリアクセスのレーテンシーが処理時間に影響する。しかし、Zバッファを階層化して、たとえば、最上位層のZバッファのブロックを256分割くらいにすれば、そのZバッファのZ値はキャッシュに格納することができるようになり、キャッシュ内でZ値の判定を行うことができるので、Z値参照によるレーテンシーを減らし、高速な処理が可能となる。

#### 【0109】

描画処理部46は、Zカリング処理を有効ならしめるため、手前のオブジェクトを含むブリックから描画処理を開始して、段階的に後方に位置するオブジェクトを含むブリックを描画処理するように、スケジューリングを決定し、処理手順リストを作成する。描画処理部46は、前方のブリックの描画処理が終わると、イベントを起動し、その前方ブリックについて階層化Zバッファを作成し、上位層のZバッファのデータを統合部48に転送する。統合部48は、転送された上位層のZバッファのZ値と、描画処理を予定して

いる後方のブリックに含まれるオブジェクトのZ値を比較して、オブジェクト間の前後関係を判定し、後方のブリックに含まれるオブジェクトの描画処理が必要かどうかを判断する。

#### 【0110】

描画処理部46は、前方の複数のブリックの描画処理が終わった時点で、その複数のブリックについて階層化Zバッファを作成して、それらのブリックの上位層のZバッファのデータを統合部48に転送してもよい。これにより、統合部48は、前方の複数のオブジェクトに隠れる後方オブジェクトをカリングすることができる。このように、描画処理部46は、階層化Zバッファの統合部48への転送タイミングをブリック単位で調整することにより、ある程度の数の前方ブリックの描画処理が終わってから、後方ブリックのカリング処理が実行されるように、カリング処理の実行タイミングを明示的に決めることができる。また、描画処理部46は、比較的サイズの大きいオブジェクトを含む前方ブリックだけを選択して、その選択された前方ブリックの階層化Zバッファを統合部48へ転送するなど、カリング処理に利用する前方ブリックを選択することにより、どの前方ブリックを用いてどの後方ブリックをカリング処理するかを指定することもできる。このようにカリング処理の戦略をブリック単位で明示的に決めることにより、カリング処理の柔軟性を高めることができる。

#### 【0111】

図28、図29は、それぞれ図27の最下位層のZバッファ250、最上位層のZバッファ254のZ値の例を説明する図である。図28におけるZ値は、描画処理部46が処理するブリック内で局所的に規定された深さであり、オブジェクトが近くにあるほどZ値は小さく、オブジェクトが最も手前にある場合はZ値はゼロである。Zバッファ法では、Z値を最大値 $Z_{max}$ で初期化しておき、各ピクセルについてもっとも手前にあるオブジェクトのZ値を各ピクセルの最終的なZ値として求める。したがって、求められたZ値が最大値 $Z_{max}$ のままであるピクセルがあれば、そのピクセルには投影されるべきオブジェクトが存在しないことを意味する。図29の最上位層のZバッファ254における4分割された各ブロックのZ値は、図28の最下位層のZバッファ250におけるそれぞれのブロックに対応する16ピクセルのZ値の最大値で与えられる。なお、Z値は、オブジェクトが近くにあるほど大きくなり、オブジェクトが遠くにあるほど小さくなるように定義することもでき、そのような定義を用いても、以下のZカリング処理に本質的な違いは生じない。

#### 【0112】

図30は、統合部48におけるZカリング処理を説明する図である。統合部48は、描画処理部46から図29に示された最上位層のZバッファ254の値を受け取ると、Z値のダイナミックレンジを調整して、元の空間における絶対的なZ値に補正する。同図は補正後のZバッファ290の4分割された各ブロックのZ値と、これから描画処理を行う予定の後方の2つのオブジェクト280、282の関係を示している。

#### 【0113】

統合部48が判定するZカリングの条件は、後方のオブジェクト280、282のZ値の最小値がZバッファ290のZ値よりも大きいことである。Zバッファ290のZ値は、前方のオブジェクトのZ値のそのブロックにおける最大値であるから、このZカリングの条件を満たす場合は、後方のオブジェクト280、282は前方のオブジェクトの陰に隠れて見えない。たとえば、第1の後方オブジェクト280の最小Z値は、Zバッファ290の第1ブロック270のZ値よりも大きく、Zカリングの条件を満たすため、第1の後方オブジェクト280は第1ブロック270内の前方オブジェクトの陰に隠れて見えない。第2の後方オブジェクト282も同様にZカリングの条件を満たし、第2ブロック272内の前方オブジェクトの陰に隠れて見えない。一方、第3ブロック276のZ値は最大値 $Z_{max}$ をとるから、このブロック内に後方オブジェクトが存在したとしても、前述のZカリングの条件を満たすことはない。これは、第3ブロック276内に前方オブジェクトが存在していないため、このブロック内にある後方オブジェクトは視点位置から見え

ることを意味する。統合部48は、Zカーリングの条件を満たした第1の後方オブジェクト280と第2の後方オブジェクト282をカーリングして、これらのオブジェクトの描画処理を省略する。このようにブロック単位でZ値を大小比較することにより、後方のオブジェクトの描画が不要である場合を効率よく判別して、無駄なレンダリングを省き、描画処理速度を上げることができる。

#### 【0114】

ブロック単位のカーリング処理の具体的な例を説明する。図31(a)に示すように、視点Oから見た場合の近景にはメニュー302と2つのB-box 304、306があり、遠景に4つのB-box 308、309、310、312が存在し、これらのオブジェクトを投影面300に投影して描画する。図31(b)は、図31(a)の上面図であり、同図に示すように、近景の3つのオブジェクト302、304、306は、距離Z1の位置にある前方クリッピング面と距離Z2の位置にある後方クリッピング面に挟まれた前方ブロック320内にあり、遠景の4つのオブジェクト308、309、310、312は、距離Z3の位置にある前方クリッピング面と距離Z4の位置にある後方クリッピング面に挟まれた3つに分割された後方ブロック322a、322b、322c(以下、後方ブロック322と総称することがある)内にある。

#### 【0115】

前方ブロック320、後方ブロック322の順に描画処理がスケジューリングされ、前方ブロック320の描画処理後に前方ブロック320の階層化Zバッファにおける上位層のZバッファの情報が後方ブロック322の描画処理の際に利用され、描画処理の不要なオブジェクトまたはブロックがカーリングされる。

#### 【0116】

図31(a)、(b)に示すように、視点Oからメニュー320を見た場合のビューボリューム324には、いずれの遠景のB-box 308、309、310、312も含まれない。一方、図32(a)、(b)に示すように、近景のB-box 304のビューボリューム326には、遠景のB-box 310が含まれる。また、図33(a)、(b)に示すように、近景のもう一つのB-box 306のビューボリューム328には、他の遠景のB-box 312が含まれる。したがって、これら2つの遠景のB-box 310、312はカーリングの対象となり、これらのB-box 310、312のそれぞれを含む2つの後方ブロック322a、322cについては描画処理が行われない。

#### 【0117】

なお、ここでは、ブロック全体がカーリングされる例を説明したが、ブロック内のB-boxが個別にカーリングされてもよい。たとえば、後方ブロック322が3つに分割されてしまう、一つのブロックである場合、ブロック全体がカーリングされるのではなく、ブロック内の2つの遠景のB-box 310、312が個別にカーリングされて描画処理の対象外となった後に、残りのB-box 308、309について描画処理がなされる。さらに、カーリング処理は、B-boxレベルで行う他、B-box内のオブジェクトのポリゴンモデルのレベルで行ってもよい。ブロック内のオブジェクトの特性やそのブロックを処理するレンダリングユニットのCPU性能やメモリの容量によって、B-boxレベルのカーリング、ポリゴンレベルのカーリングのいずれかを選択してもよく、ブロックごとにカーリング処理の戦略を最適化することができる。

#### 【0118】

さらに別の描画戦略の例として、ブロックごとに隠面消去の手法を異ならせ、たとえば近景のブロックに対しては、計算の複雑なレイトレーシングを局所的に適用し、それ以外の外部のブロックに対してはリフレクションマッピングを用いるパーシャルレイトレーシングを採用してもよい。リフレクションマッピングでは、描画対象のオブジェクトのまわりに大きな仮想球または仮想立方体を置き、その内面にあらかじめ環境のテクスチャが貼り付けておく。そして、視線と交差するオブジェクトの交点を求め、さらにその点から反射ベクトルを延ばして仮想球または仮想立方体との交点を求め、その交点での仮想球または仮想立方体の色をオブジェクトの色としてマッピングする。レイトレーシングでもリフ

レクションマッピングでも、環境の情報を無限遠まで広げると、計算量が多くなり、メモリ効率も悪くなるため、ブリック単位で環境を限定して、描画に必要な環境データを絞り込むことができる。描画に必要な環境データを他のブリックから受け取ることができるよう、ブリック間で互いに必要なデータの交換が可能なシステム構成とし、描画対象のブリックに近いブリックからはオブジェクトデータの提供を受け、遠方のブリックからは画像データの提供を受けるなど、他のブリックから提供される環境データの詳細度に差を設けてもよい。

【図面の簡単な説明】

【0119】

【図1】図1 (a)、(b)は3次元空間における四角錐台のビューボリュームの説明図である。

【図2】図2 (a)、(b)は3次元空間の四角錐台による分割例を説明する図である。

【図3】図3 (a)、(b)は3次元空間の四角錐台による別の分割例を説明する図である。

【図4】図4 (a)、(b)は3次元空間の四角錐台によるさらに別の分割例を説明する図である。

【図5】図5 (a)～(e)は、図4の空間分割における描画処理の概略を説明する図である。

【図6】実施の形態に係る画像処理装置の構成図である。

【図7】図6の画像処理装置のグラフィック処理ブロックの機能構成図である。

【図8】図7の描画処理部の並列処理による構成例を説明する図である。

【図9】図7の構成のグラフィック処理ブロックによる分割描画処理と統合処理の手順を説明するフローチャートである。

【図10】図10 (a)は入力されたオブジェクトのグループ分けを説明する図であり、図10 (b)は、グループ分けにより生成されるグループリストを説明する図である。

【図11】同一グループのオブジェクト群を包含する分割空間の例を説明する図である。

【図12】同一グループのオブジェクト群を包含する分割空間の別の例を説明する図である。

【図13】同一グループのオブジェクト群を包含する分割空間のさらに別の例を説明する図である。

【図14】分割描画処理における処理手順リストの例を説明する図である。

【図15】図15 (a)は、双1次補間による画像データの生成処理を説明する図であり、図15 (b)は、双3次補間による画像データの生成処理を説明する図である。

【図16】図16 (a)～(c)はZ付マージ描画処理を説明する図である。

【図17】同一グループのオブジェクト群を包含する分割空間の例を説明する図である。

【図18】図17の例にさらにもう一つのオブジェクトが加わった場合における同一グループのオブジェクト群を包含する分割空間の例を説明する図である。

【図19】図9の分割空間の算出手順の変形例を説明するフローチャートである。

【図20】図18のオブジェクト群を包含する分割空間の別の例を説明する図である。

【図21】実施の形態に係る分散描画処理システムの構成例を説明する図である。

【図22】図21のパーソナルコンピュータのディスプレイに表示される画像の例を説明する図である。

【図23】図22のオブジェクト群を包含する分割空間を説明する図である。

【図24】図22のオブジェクト群を包含する別の分割空間を説明する図である。

【図25】図25（a）～（c）は、移動オブジェクトにモーションブラー処理を施す場合のメモリ領域の説明図である。

【図26】図26（a）、（b）は、デフォーカス処理を行う場合のメモリ領域の説明図である。

【図27】実施の形態に係るZカーリング処理で用いられる階層化Zバッファの説明図である。

【図28】図27の最下位層のZバッファのZ値の例を説明する図である。

【図29】図27の最上位層のZバッファのZ値の例を説明する図である。

【図30】図29の最上位層のZバッファのZ値によるZカーリング処理の例を説明する図である。

【図31】ブリック単位のカーリング処理の例を説明する図である。

【図32】ブリック単位のカーリング処理の例を説明する図である。

【図33】ブリック単位のカーリング処理の例を説明する図である。

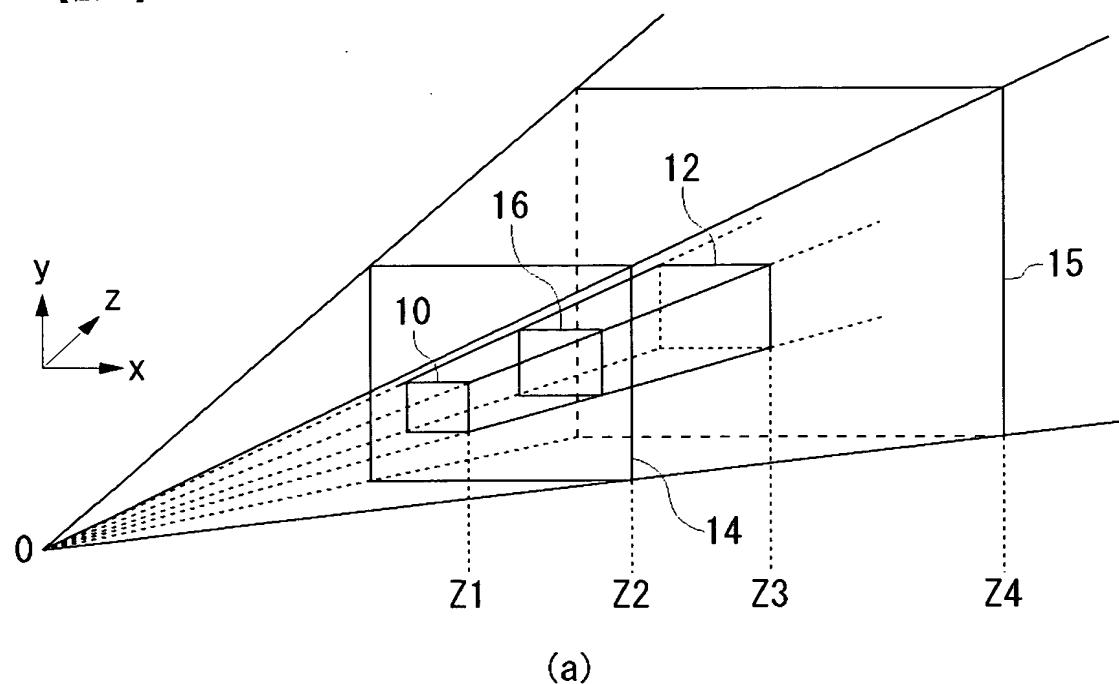
#### 【符号の説明】

##### 【0120】

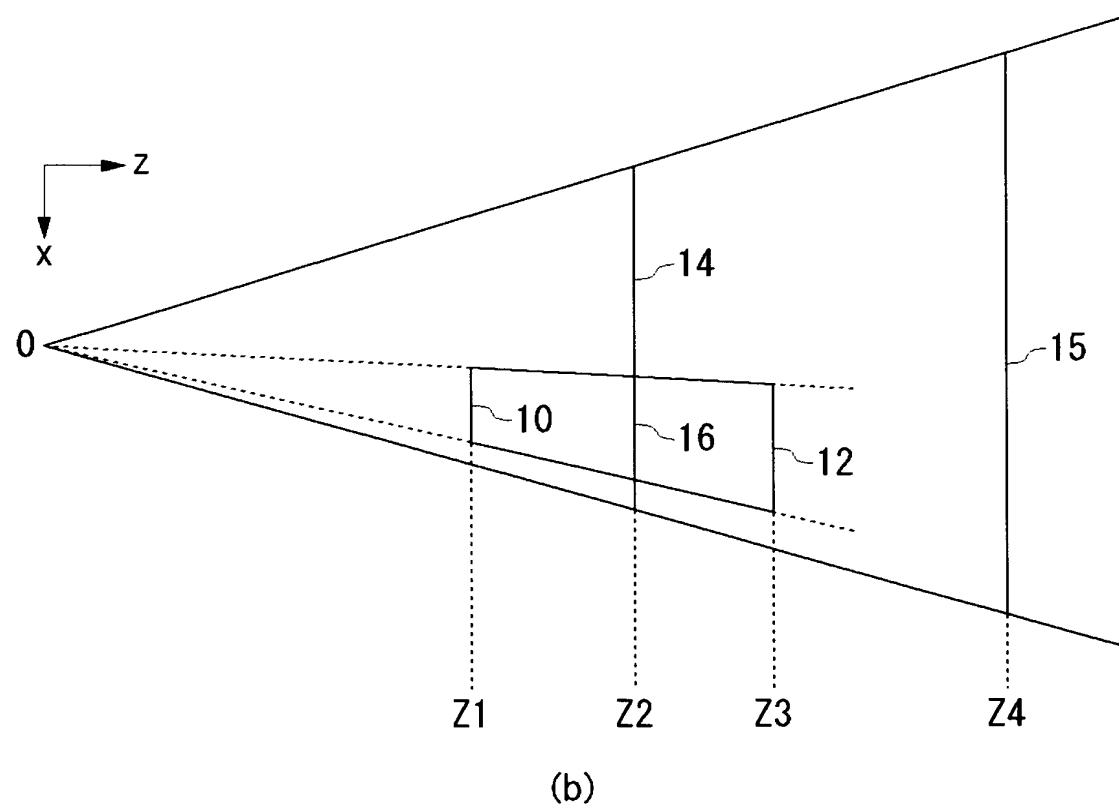
42 座標変換部、 43 レンダリング部、 44 分類部、 46 描画処理部、  
48 統合部、 50 表示部、 52 オブジェクト記憶部、 54 画像記憶部、  
100 画像処理装置、 108 I/Oポート、 110 メインCPU、 112  
メインメモリ、 114 通信部、 120 グラフィック処理ブロック、 122  
並列描画エンジン、 124 メモリインターフェイス、 126 画像メモリ。



【書類名】図面  
【図 1】



(a)

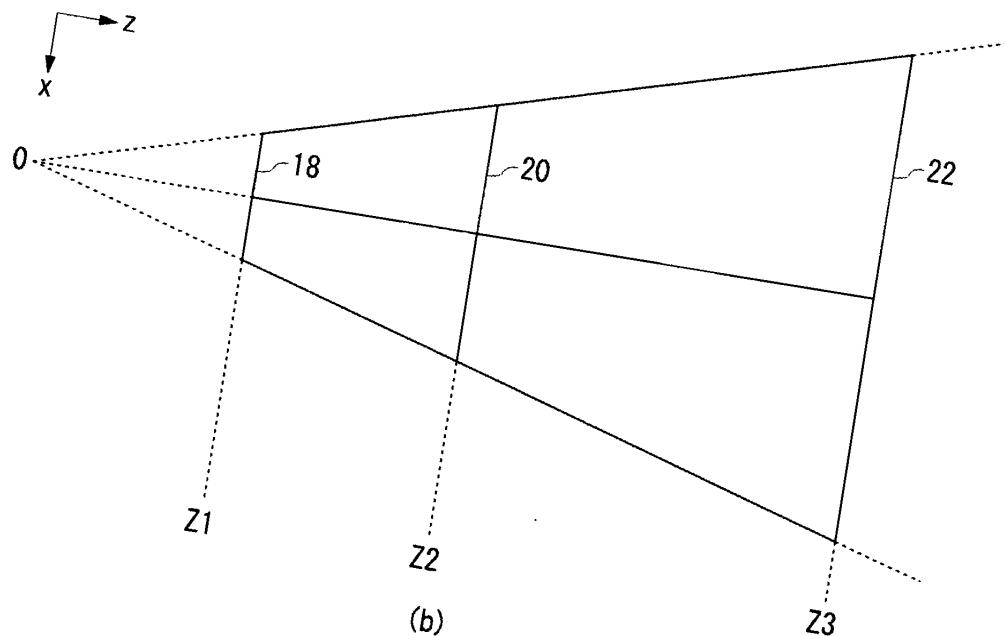
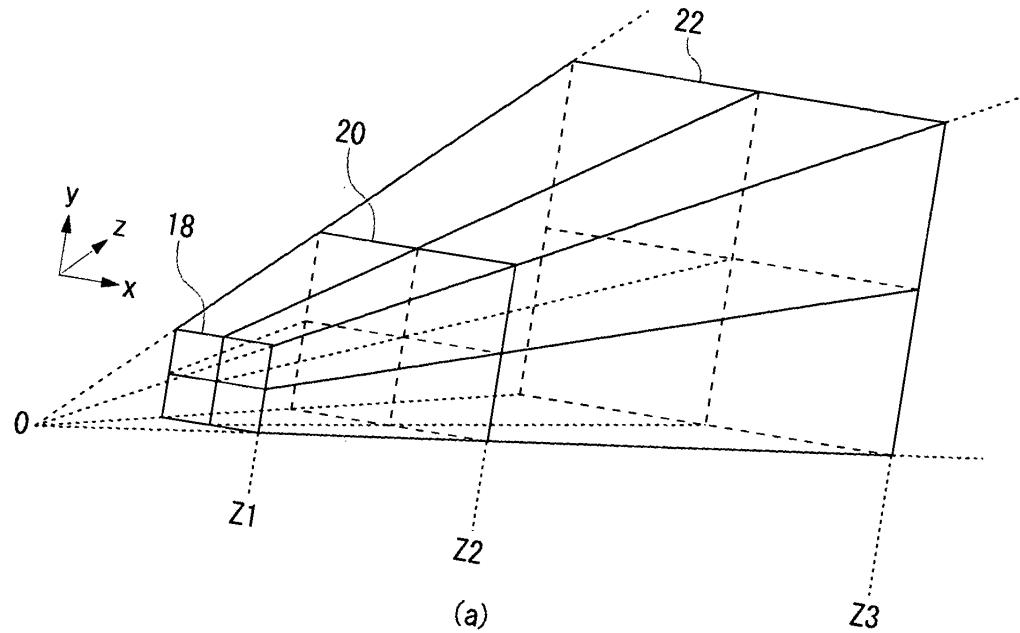


(b)

特願 2003-293804

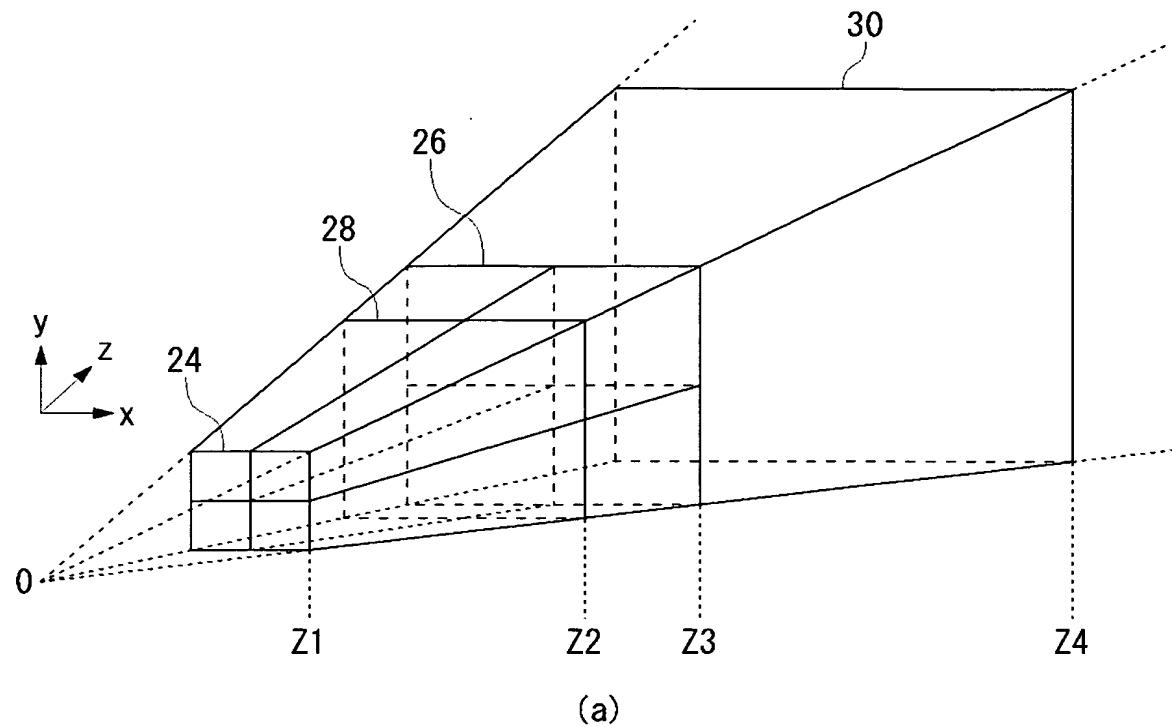
【図2】

ページ： 2/

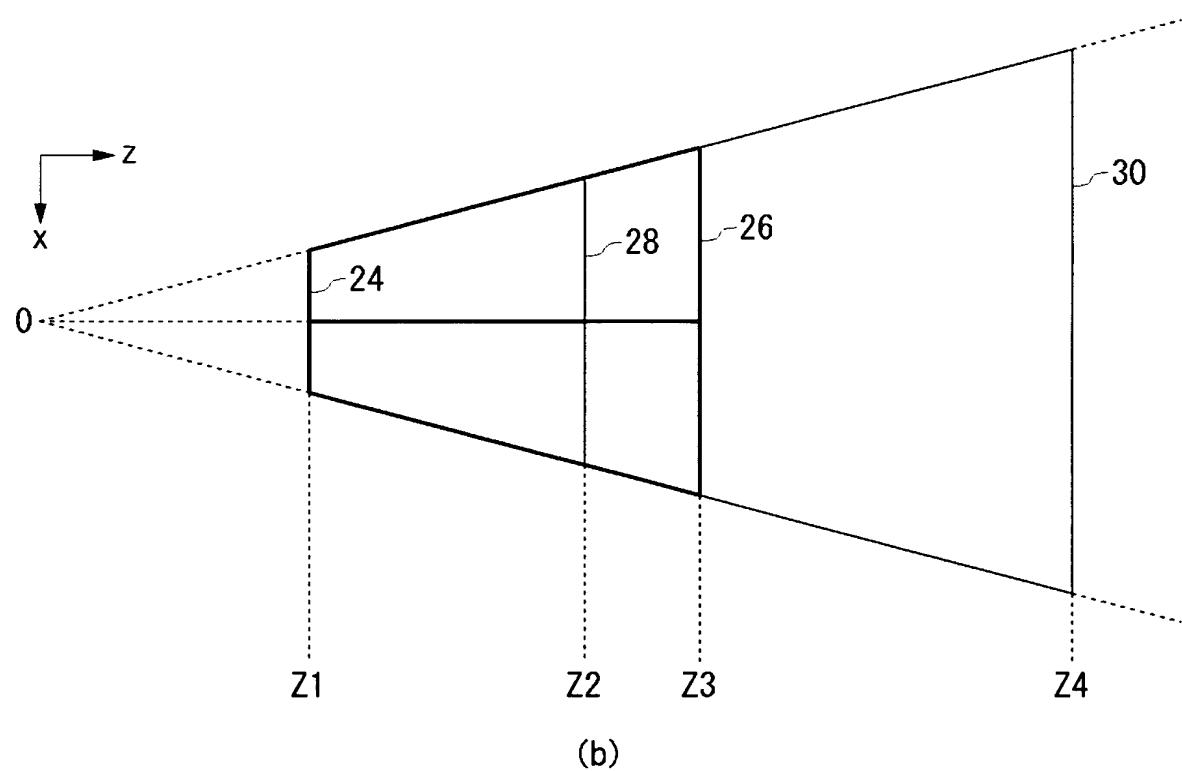


出証特 2003-3070405

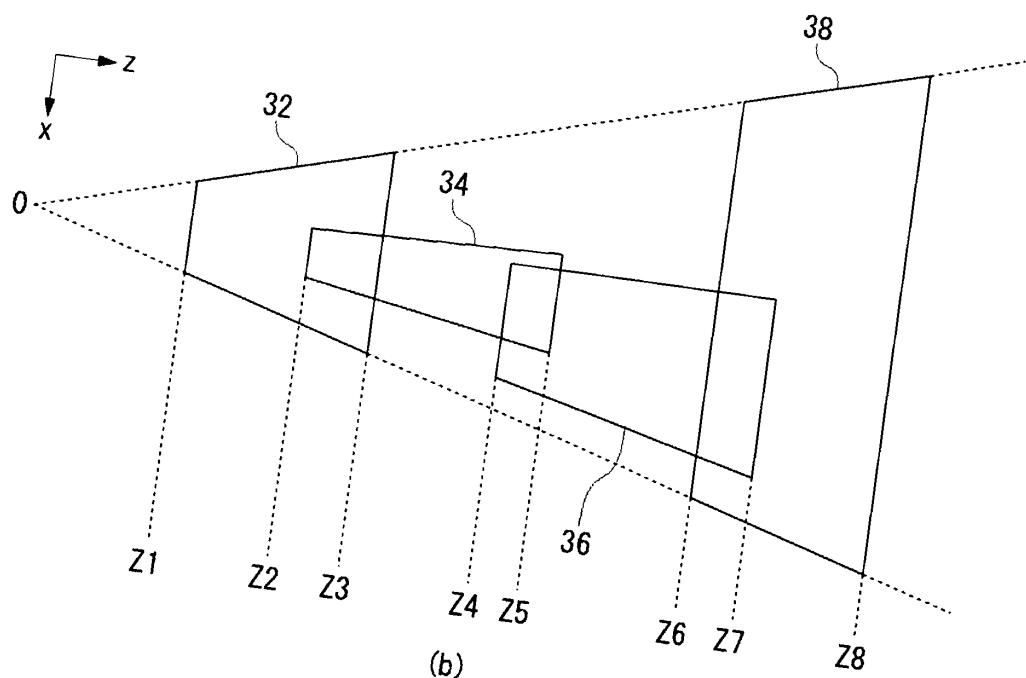
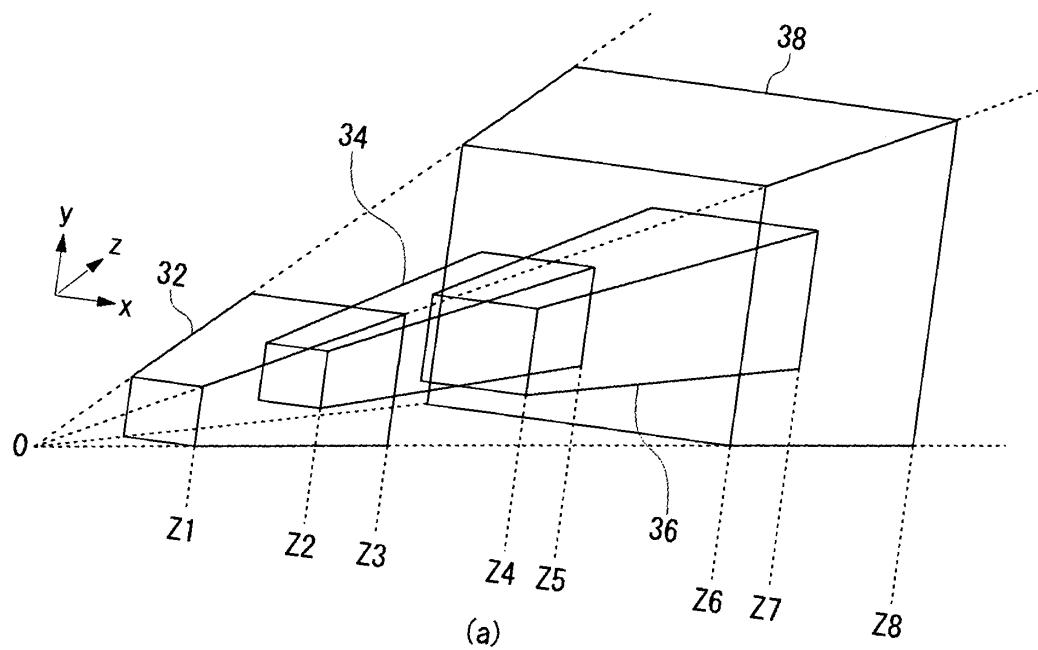
【図 3】



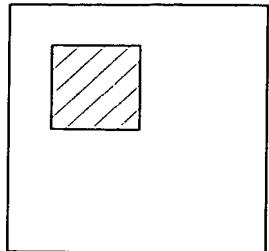
(a)



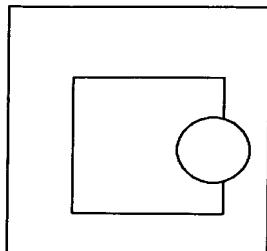
(b)



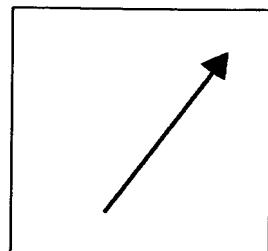
【図 5】



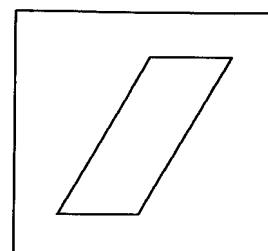
(a)



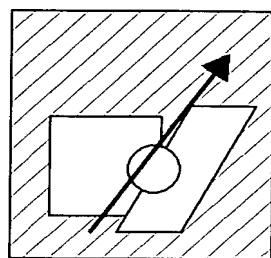
(b)



(c)

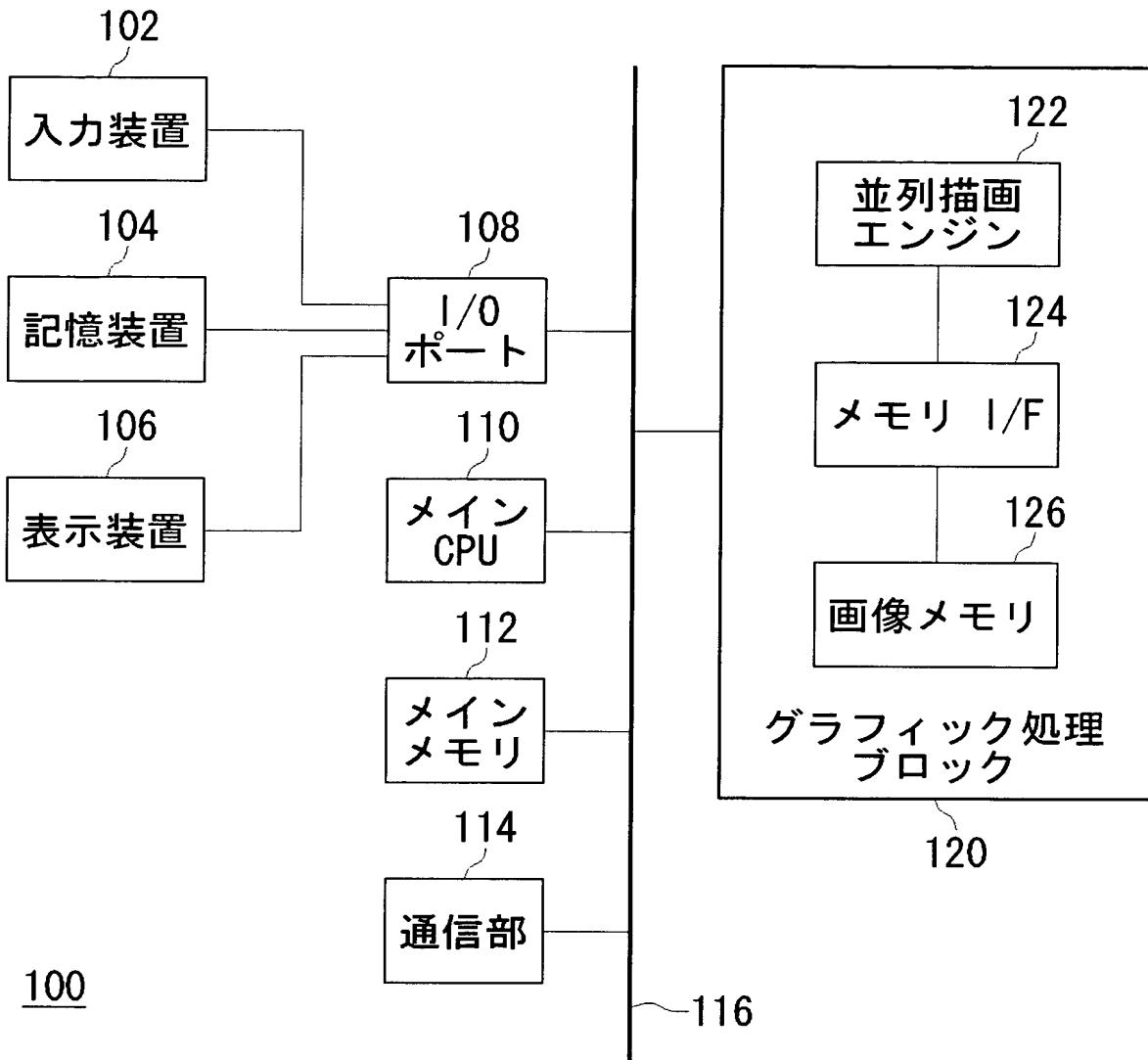


(d)

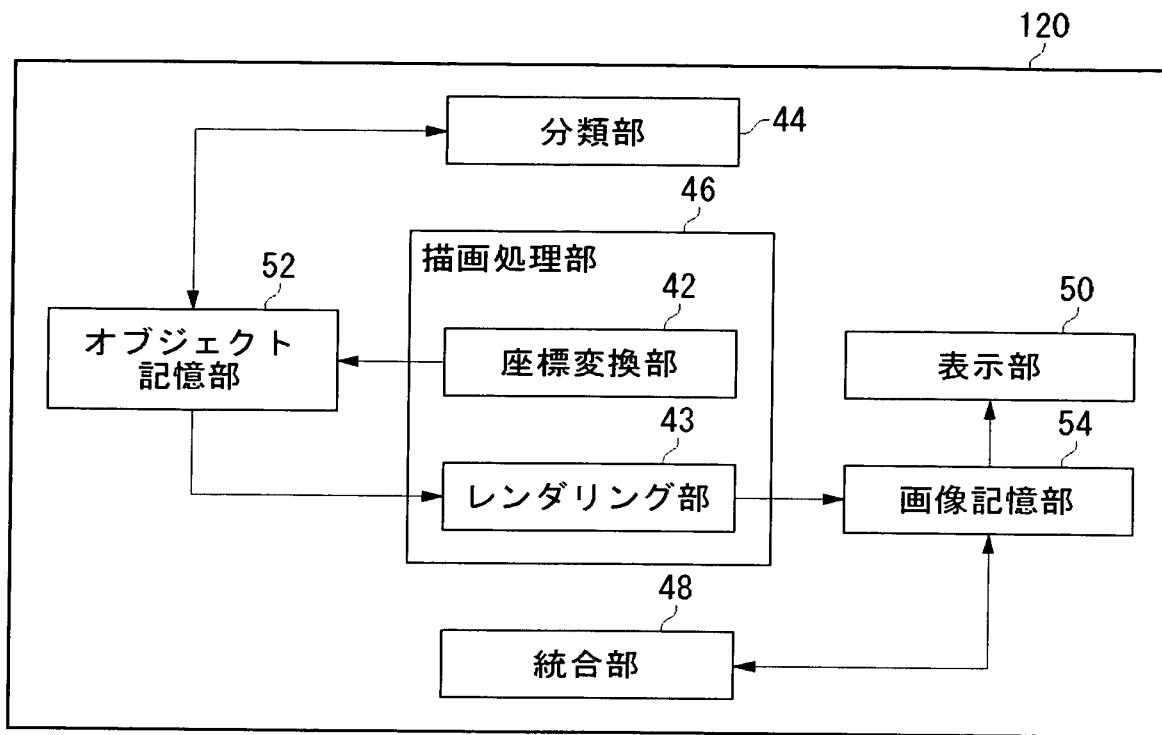


(e)

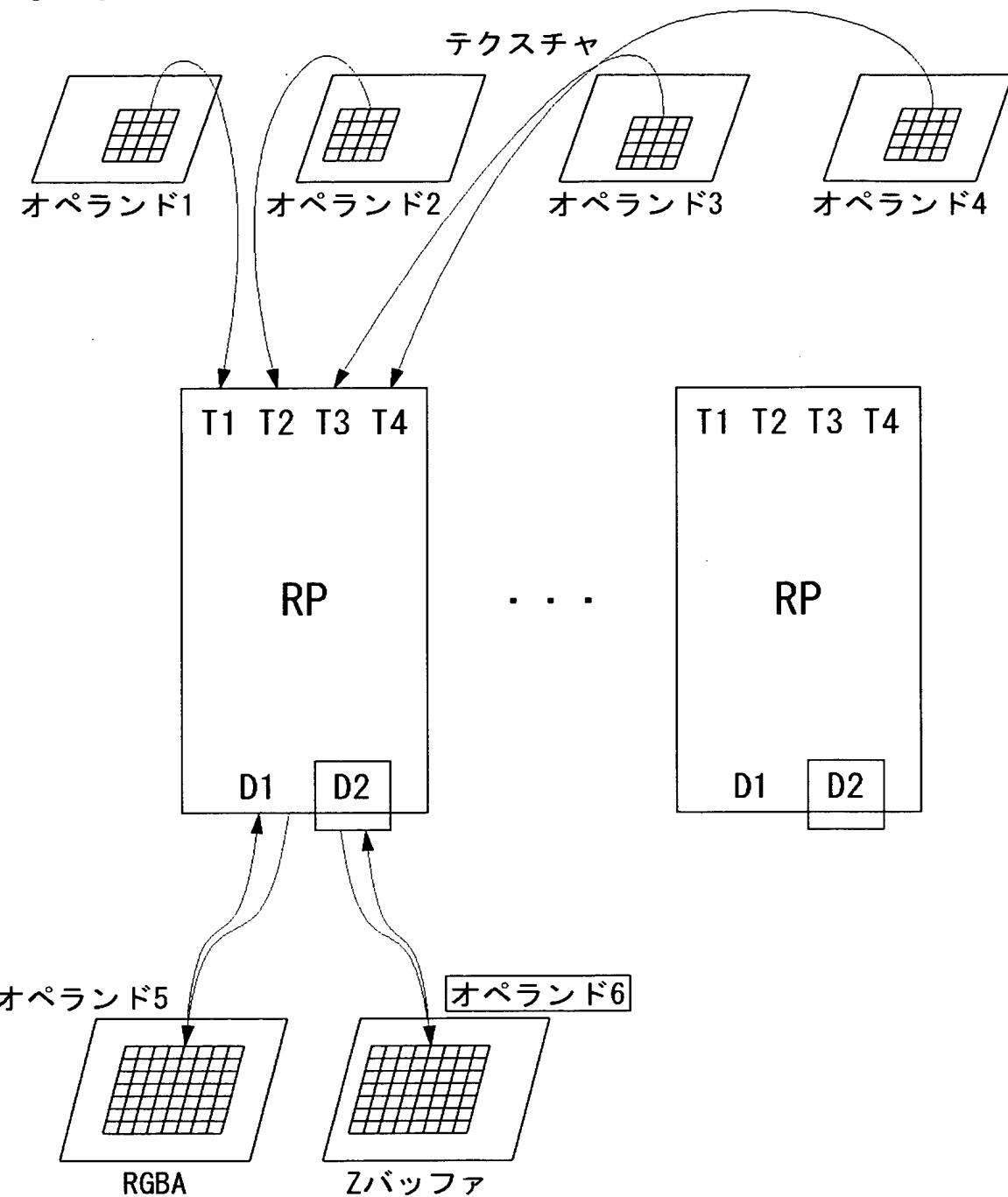
【図6】



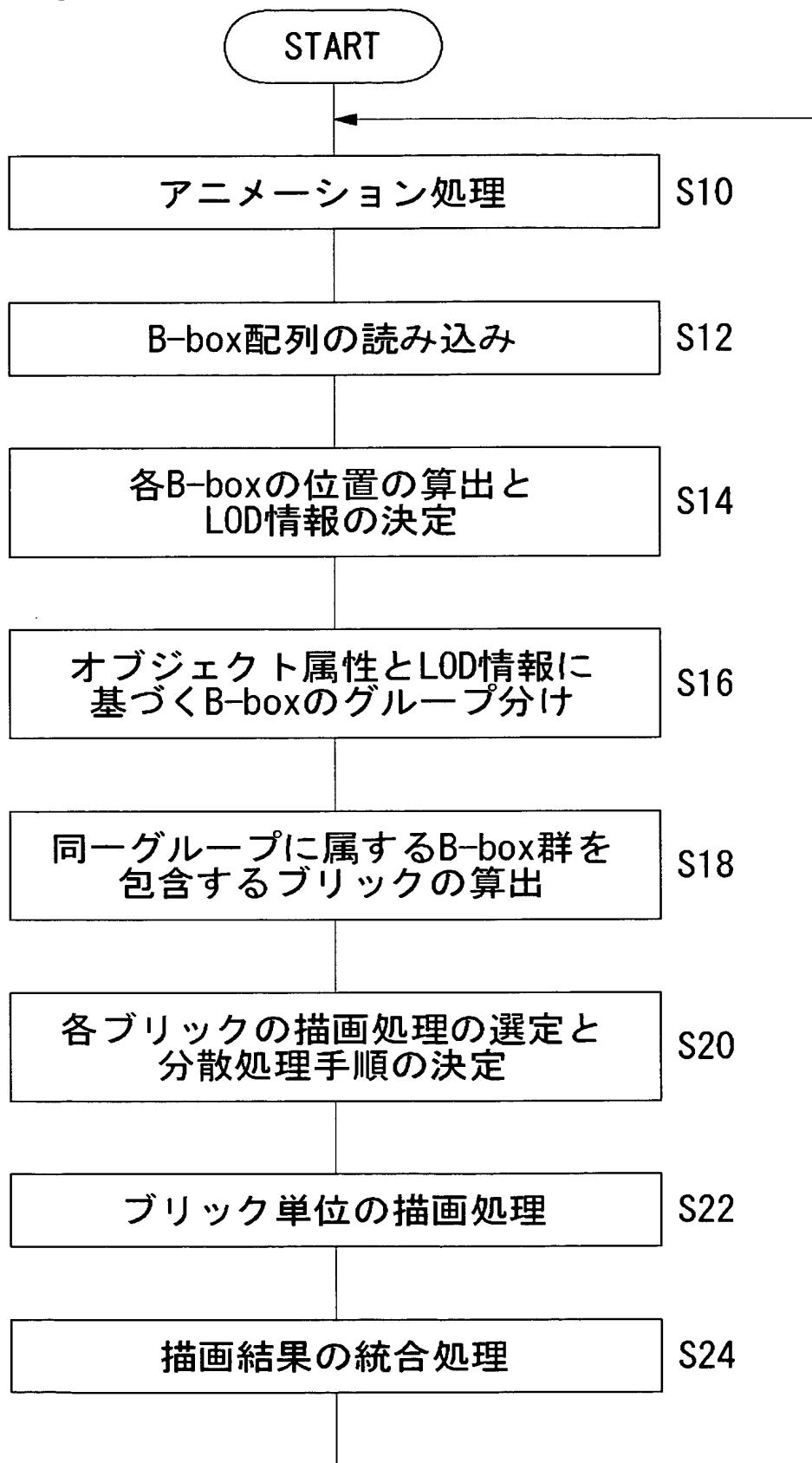
【図 7】



【図 8】



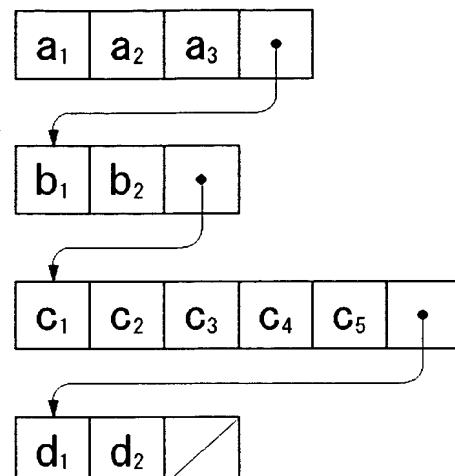
【図9】



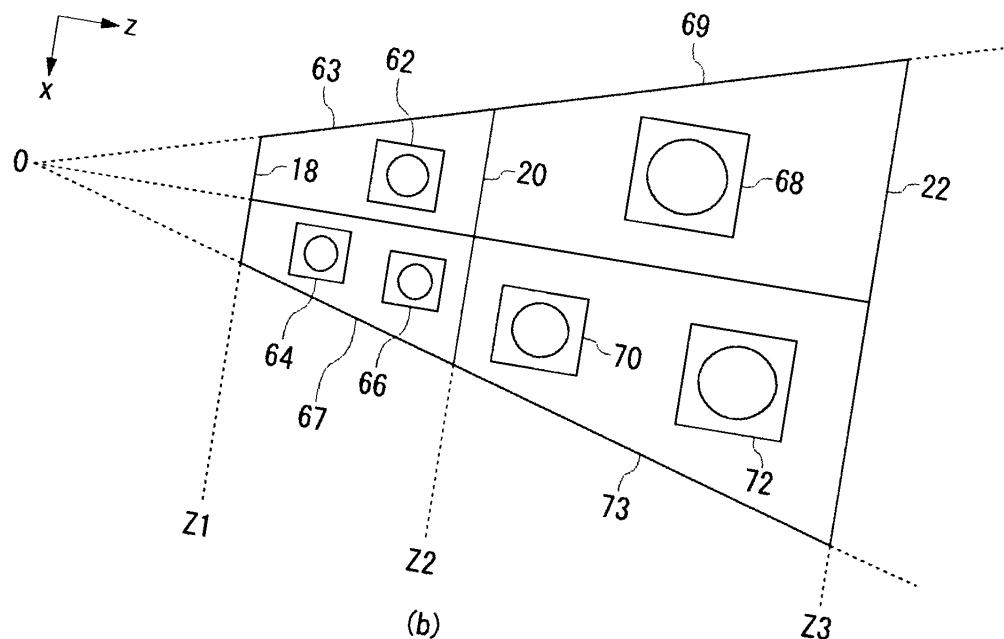
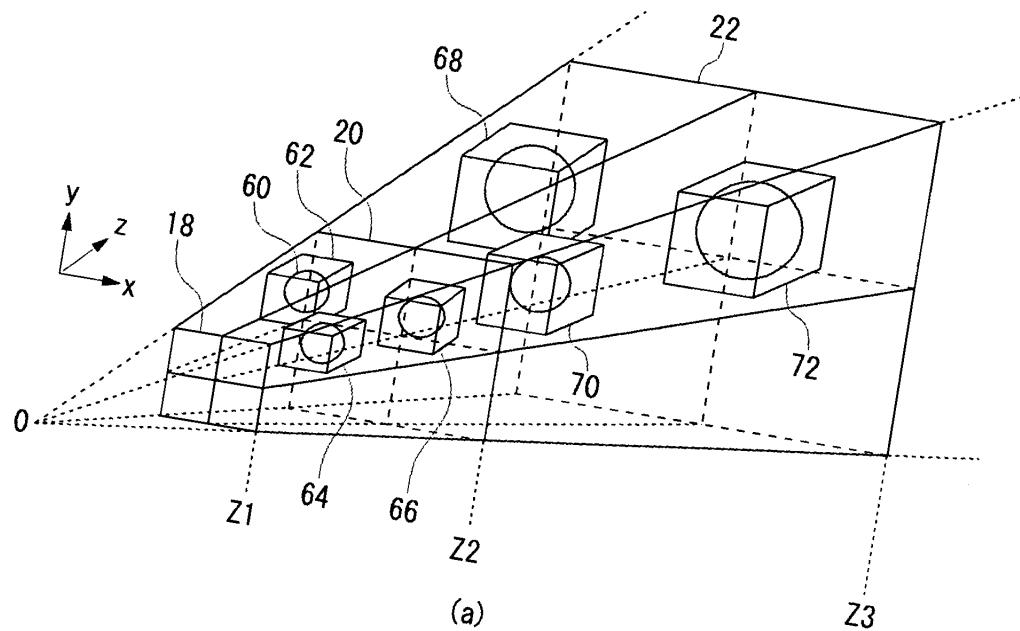
【図 10】

$c_1$	$c_2$		
$b_1$		$c_3$	$d_1$
$c_4$	$c_5$	$a_1$	$a_2$
$b_2$	$a_3$		$d_2$

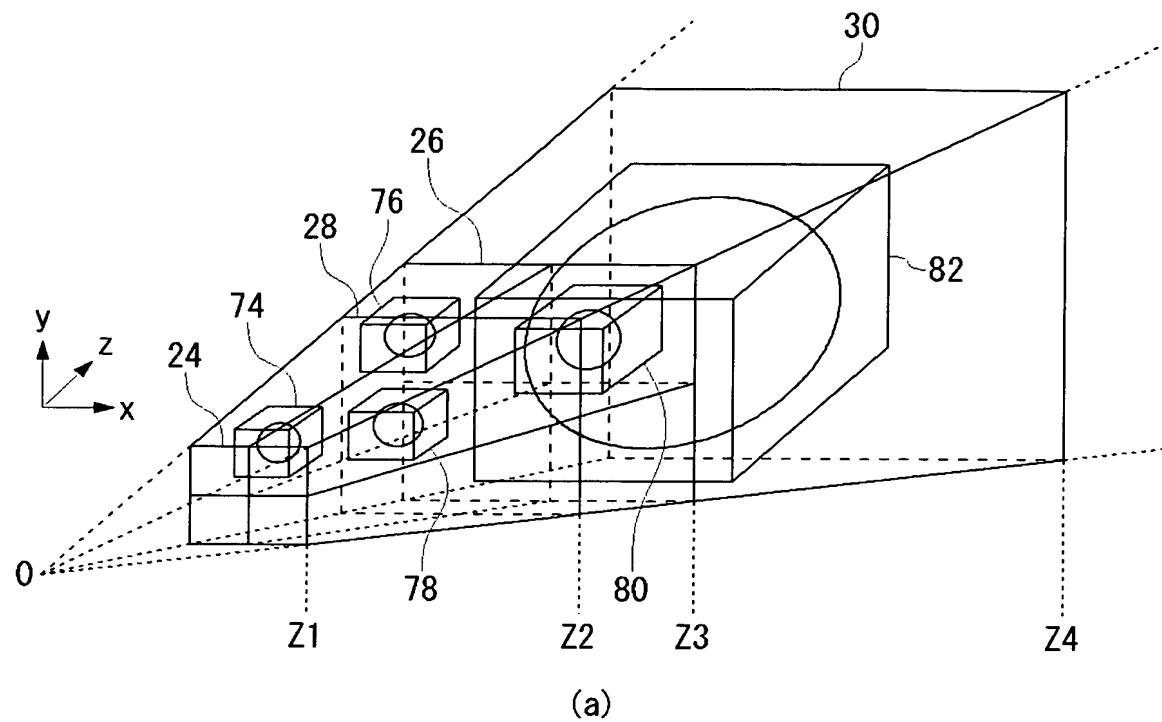
(a)



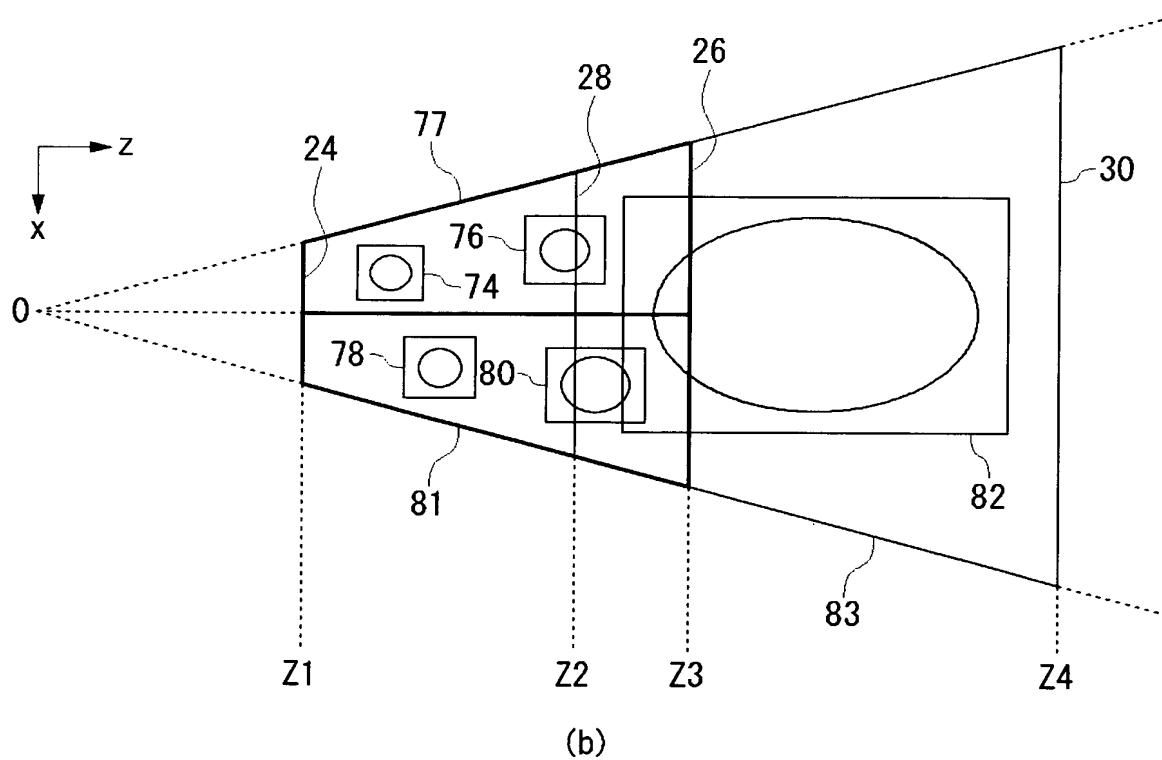
(b)



【図12】

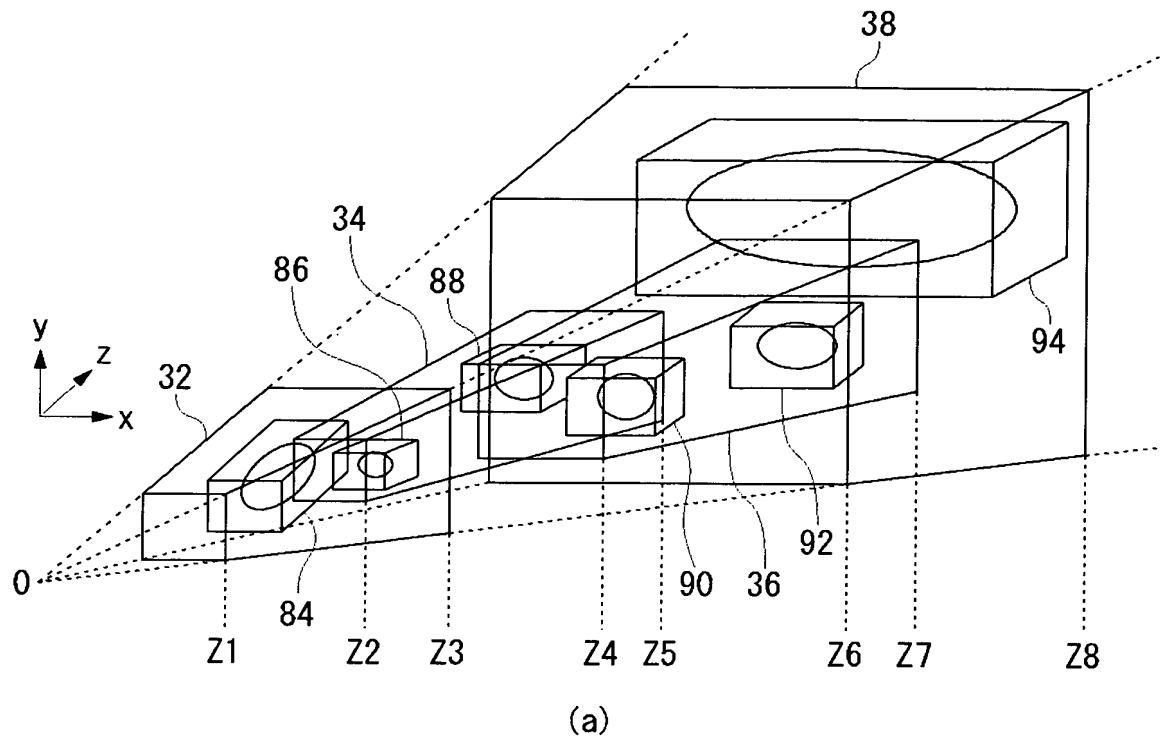


(a)

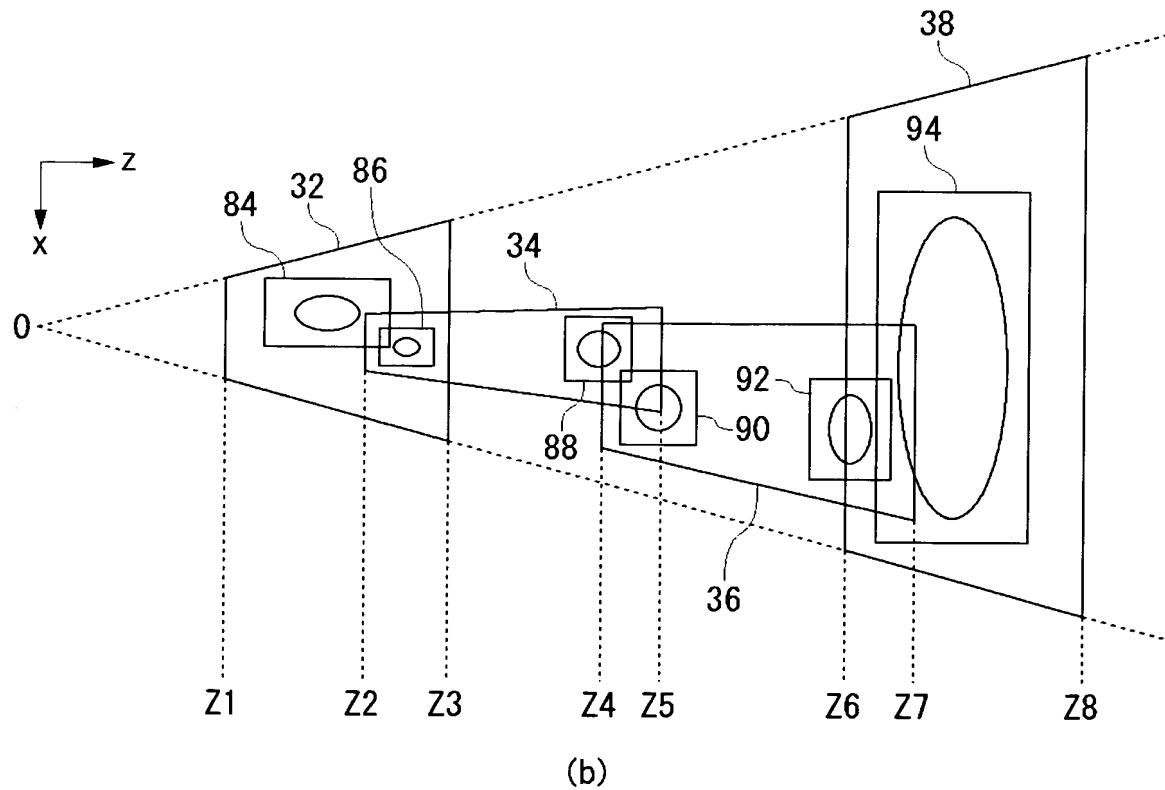


(b)

【図13】



(a)

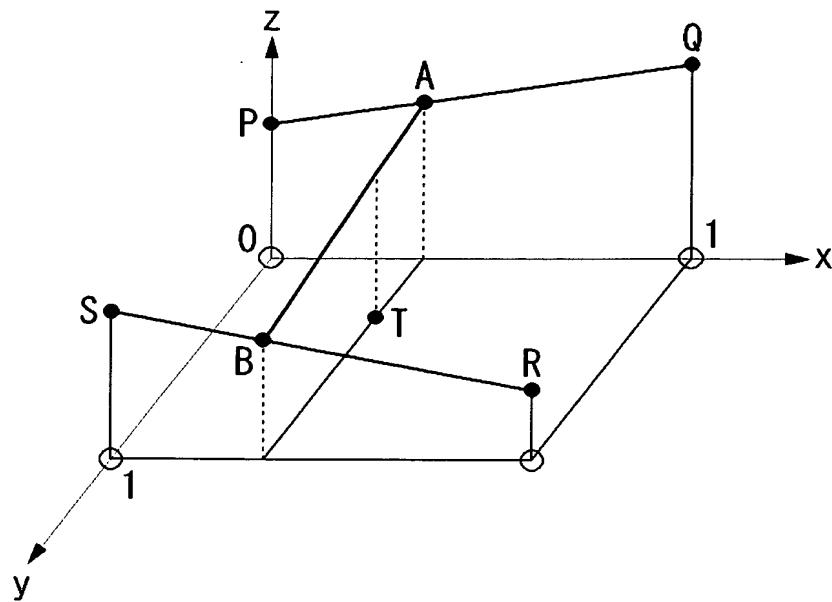


(b)

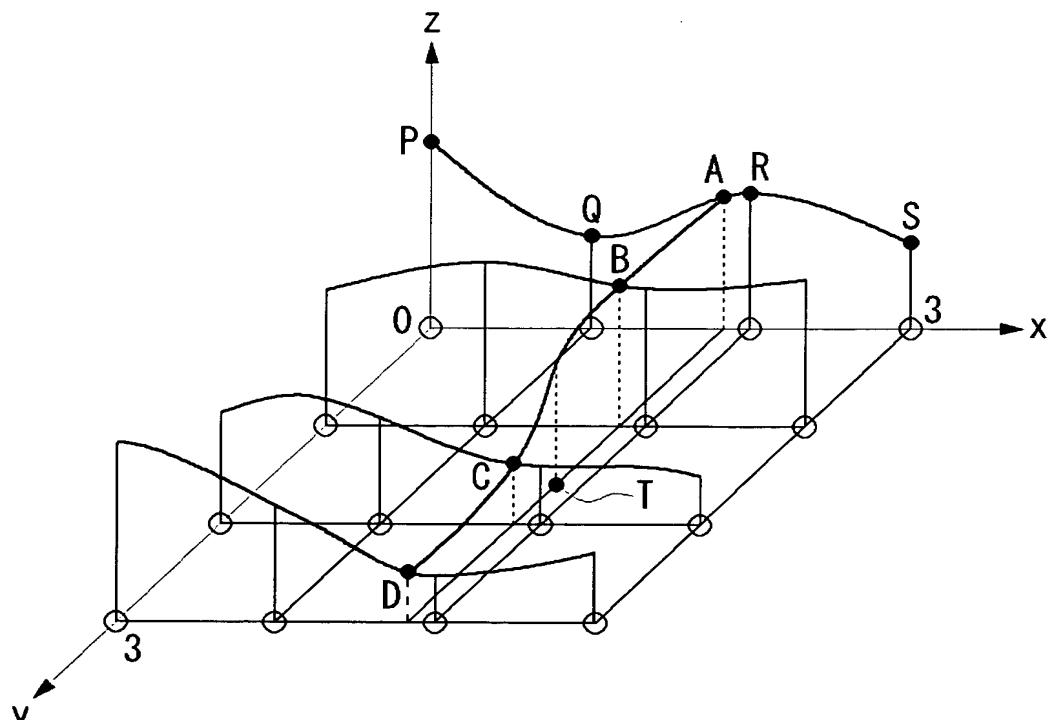
【図14】

RP1：グループA, D, G  
RP2：グループB  
RP3：グループC, E  
RP4：グループF

【図15】

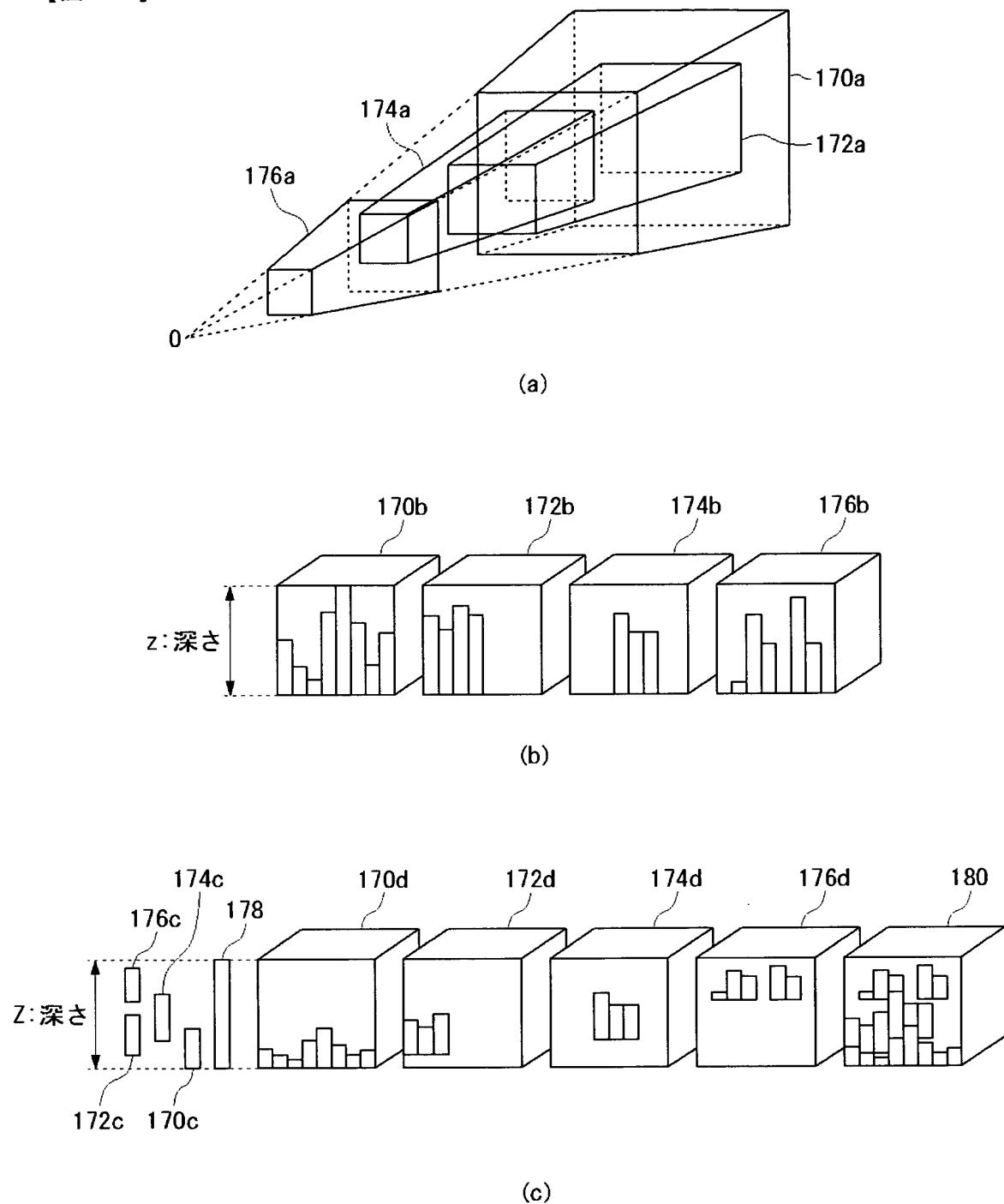


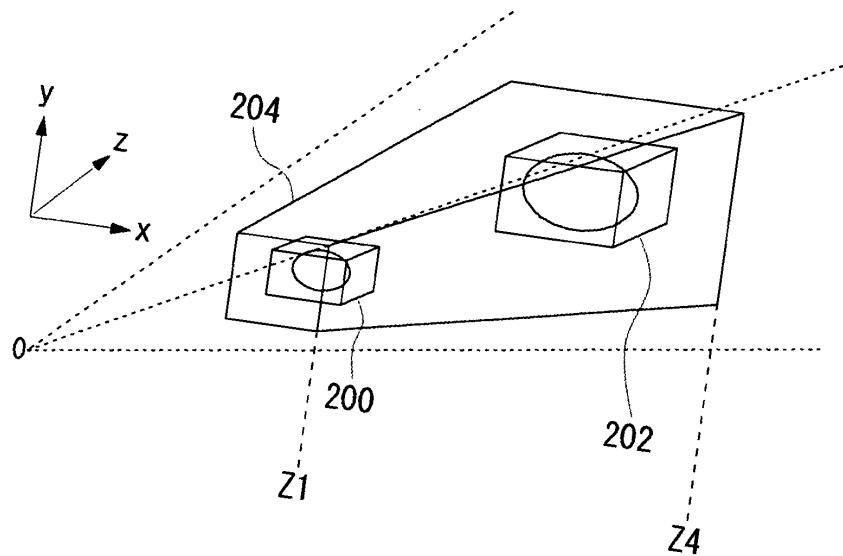
(a)



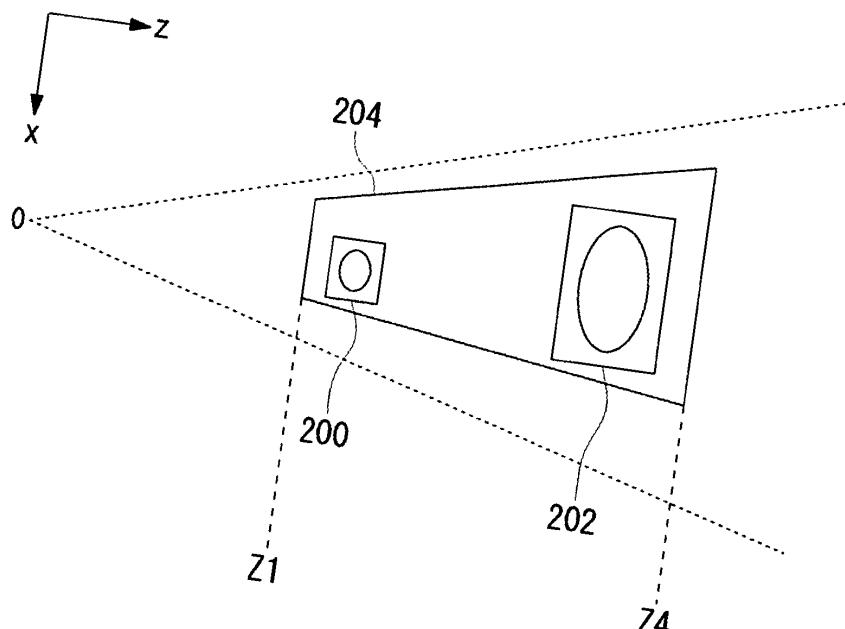
(b)

【図16】

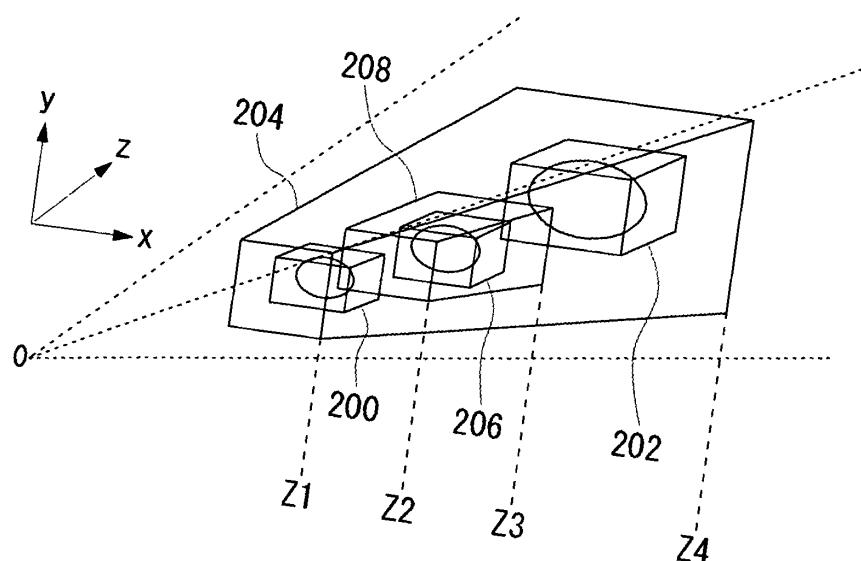




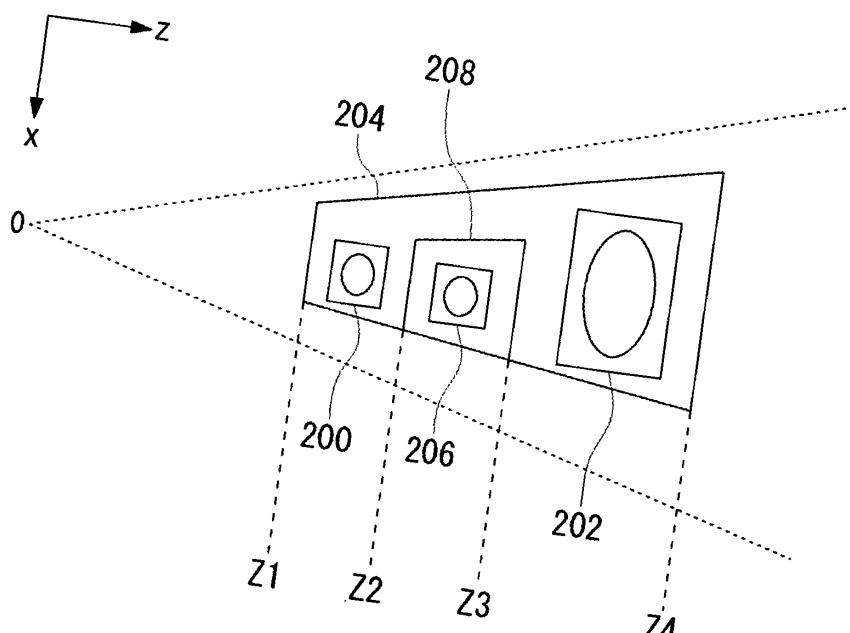
(a)



(b)

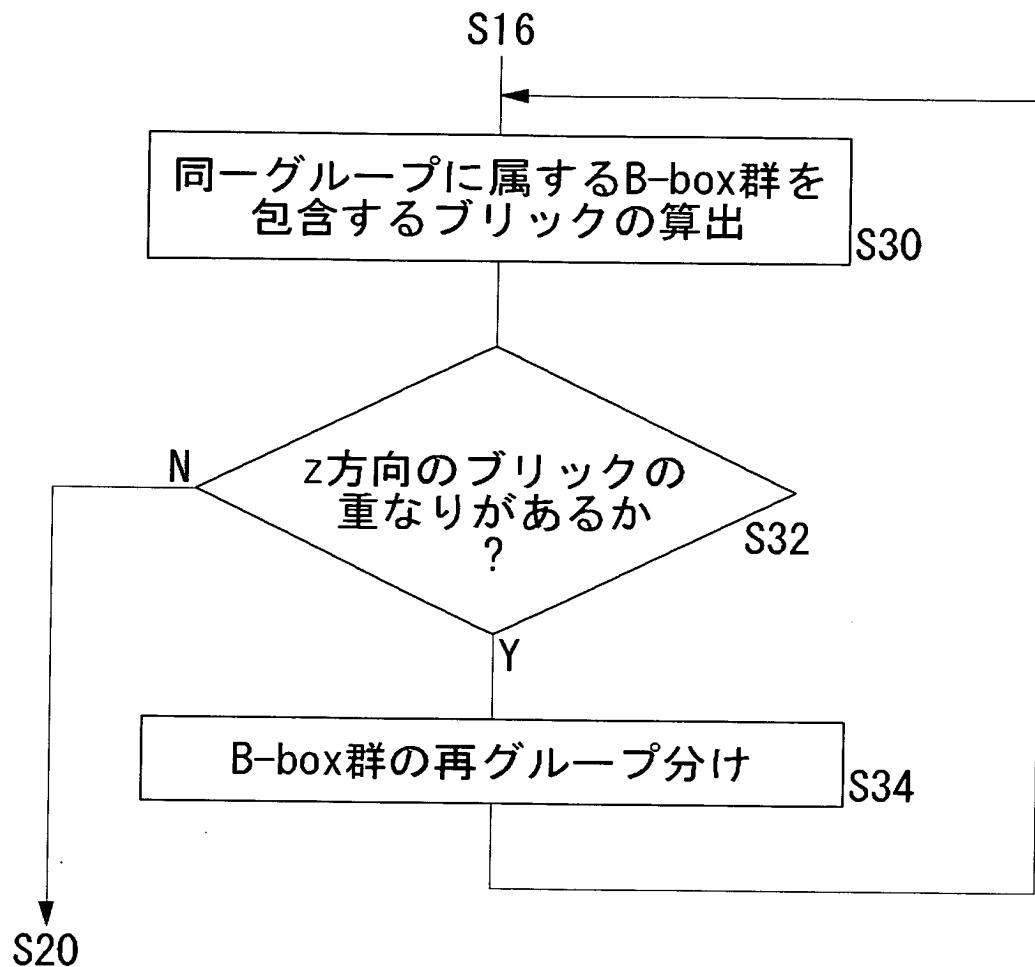


(a)



(b)

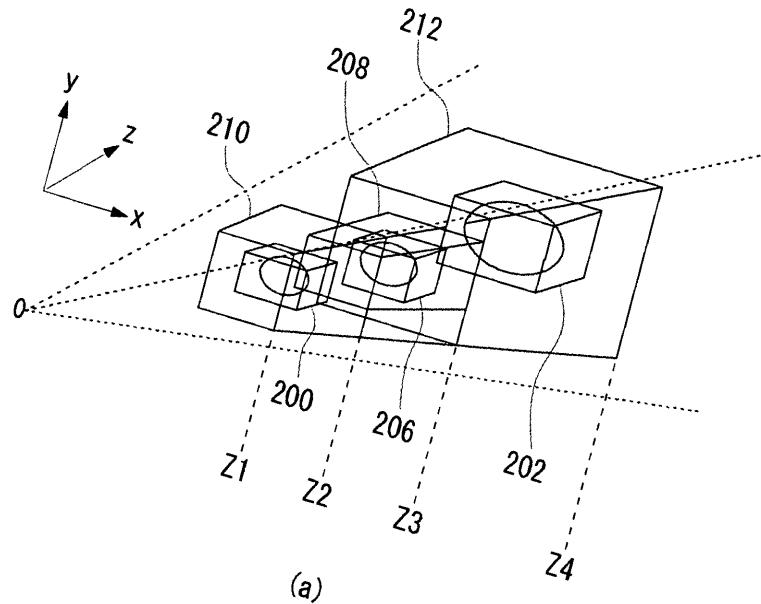
【図19】



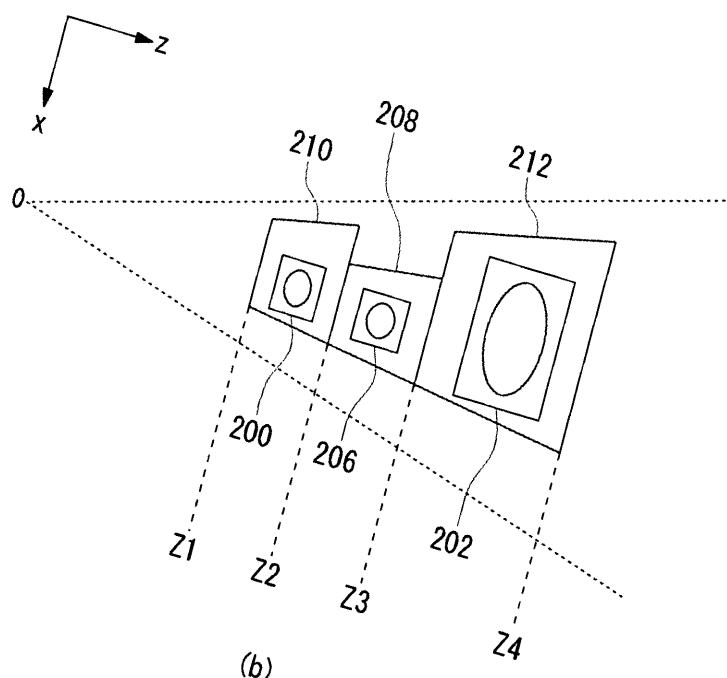
【図20】

特願2003-293804

ページ： 20/



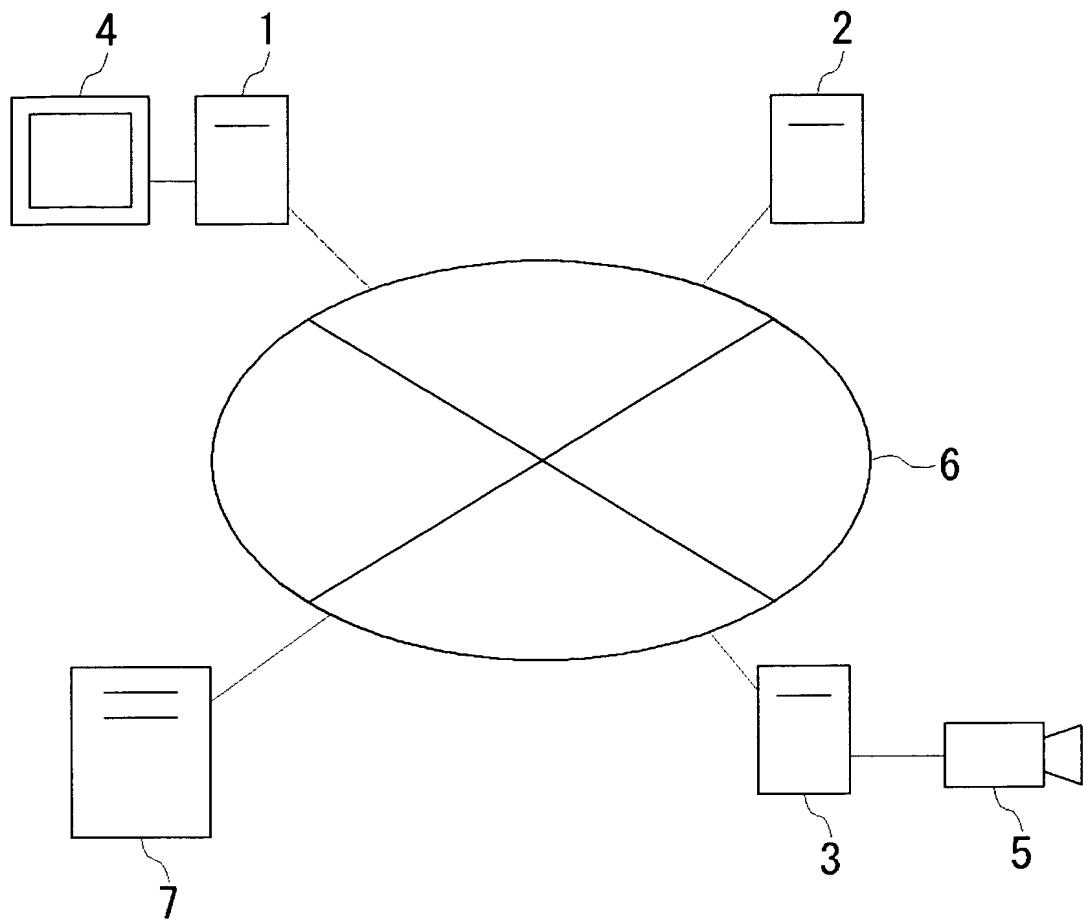
(a)



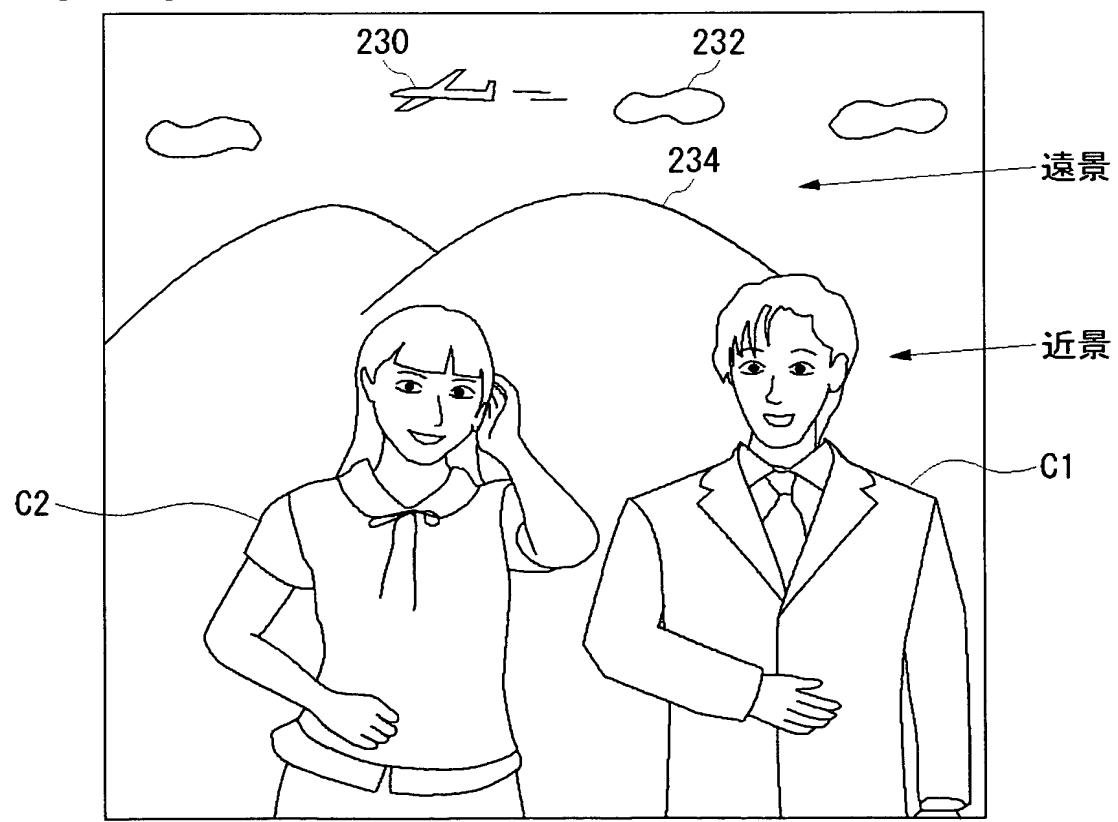
(b)

出証特2003-3070405

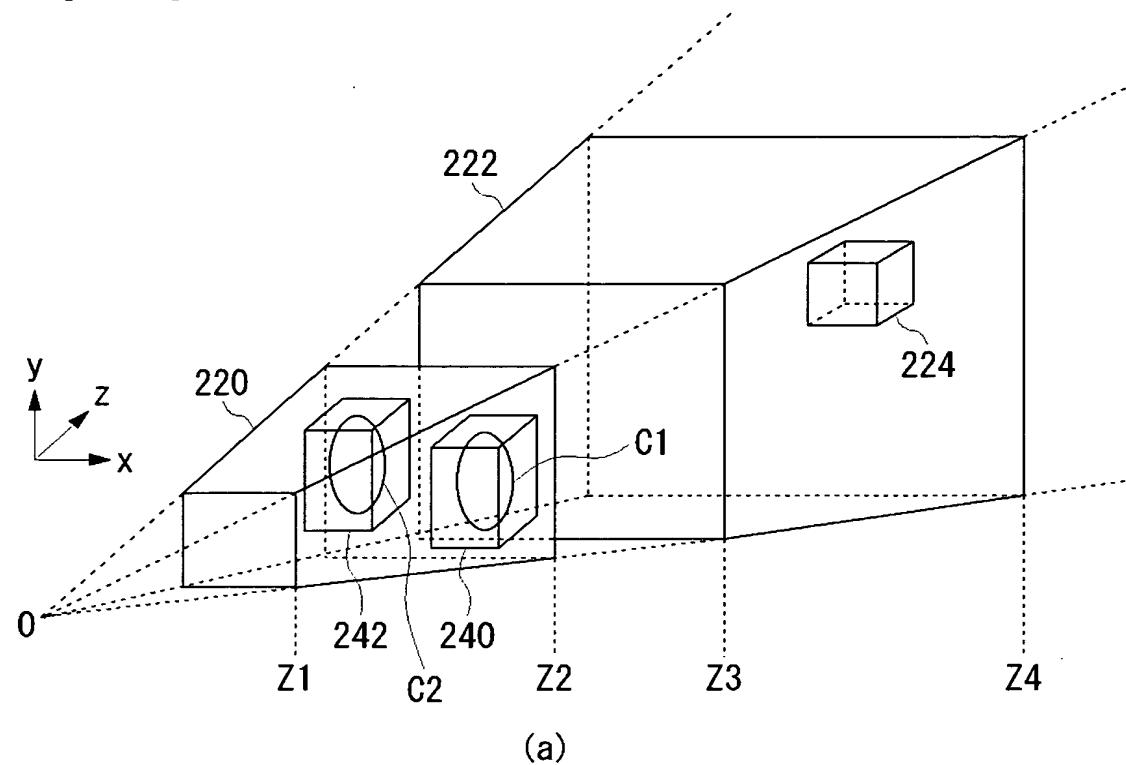
【図 2 1】



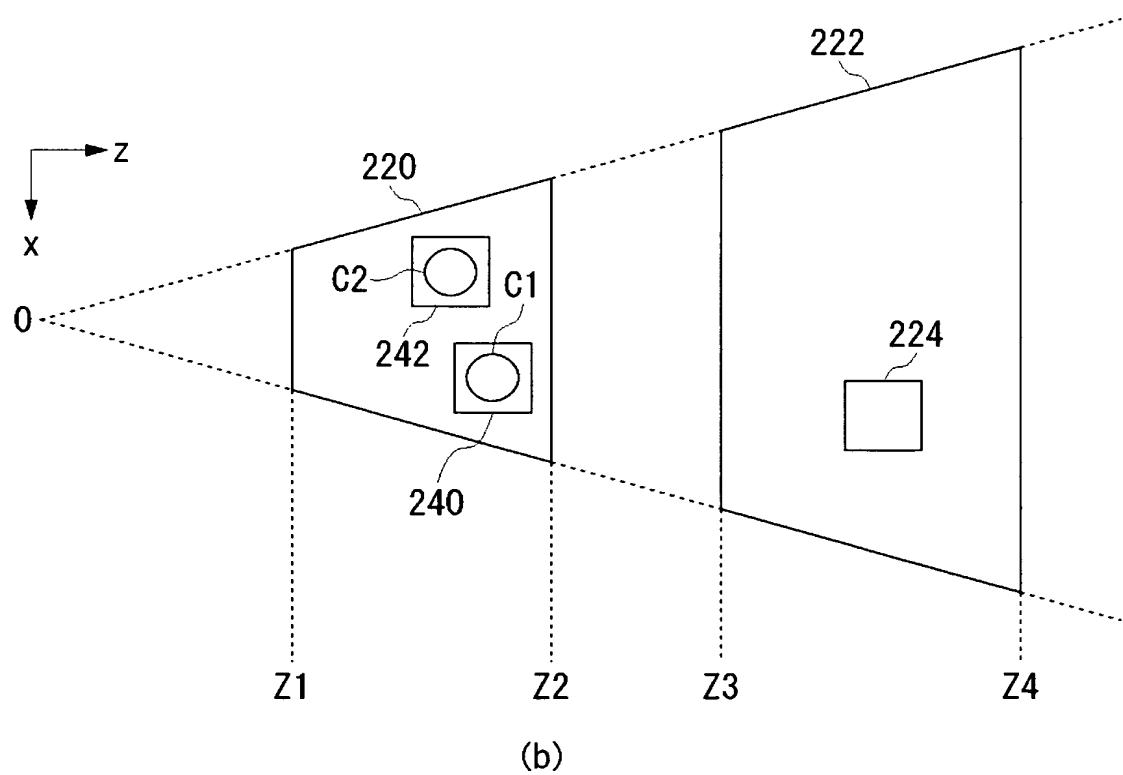
【図 2 2】



【図23】

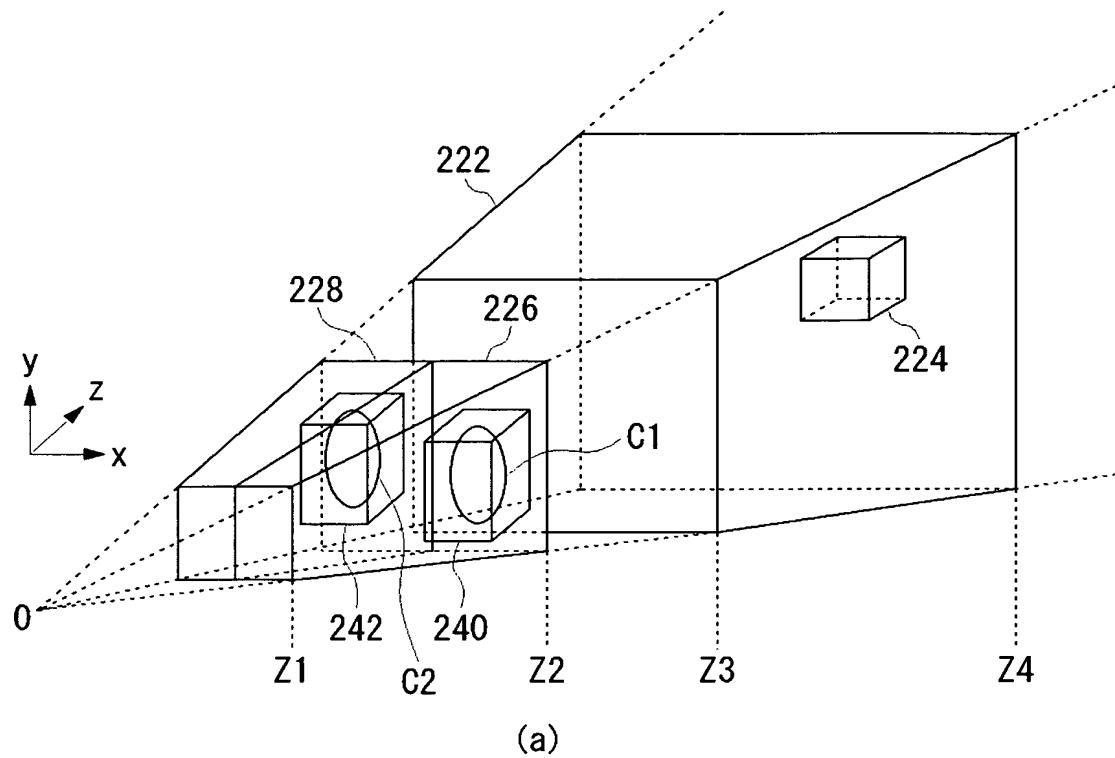


(a)

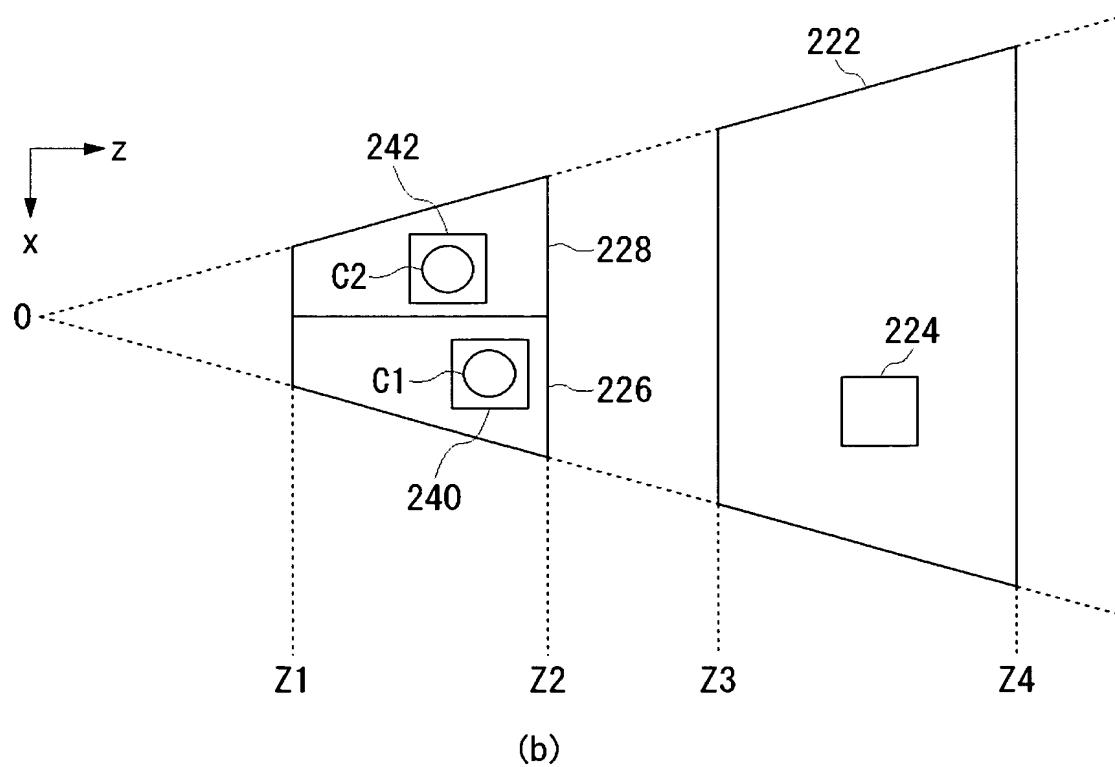


(b)

【図24】



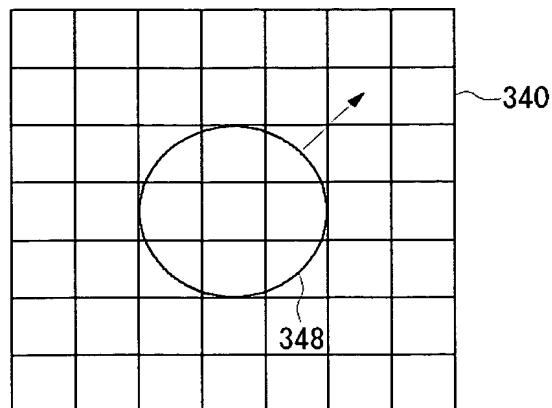
(a)



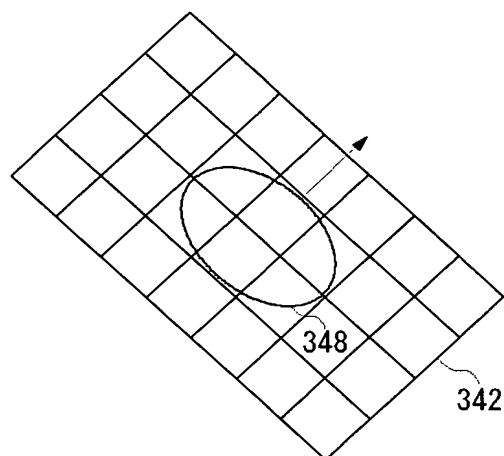
(b)

【図 25】

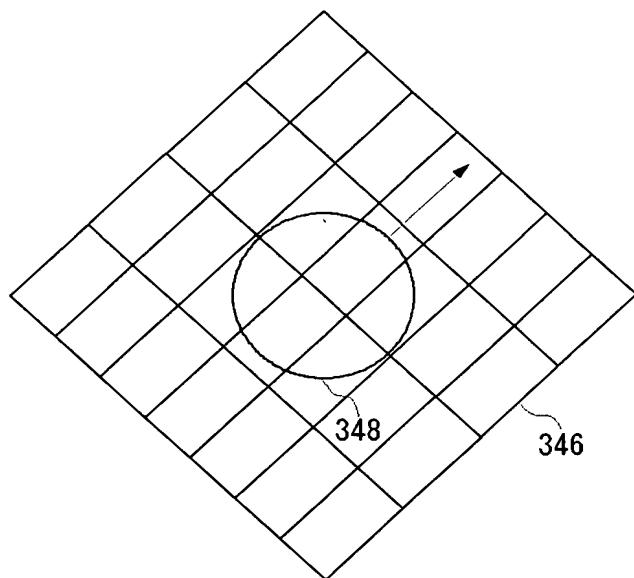
(a)



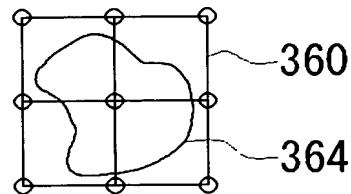
(b)



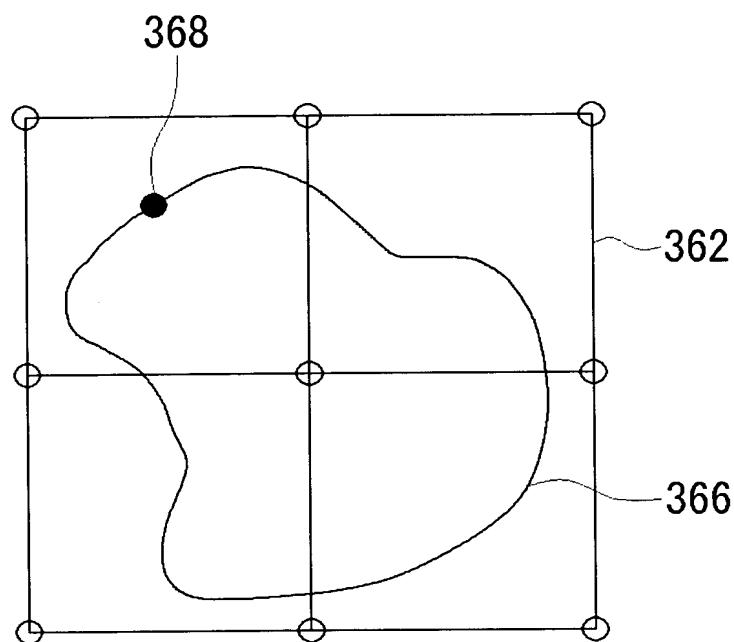
(c)



【図26】

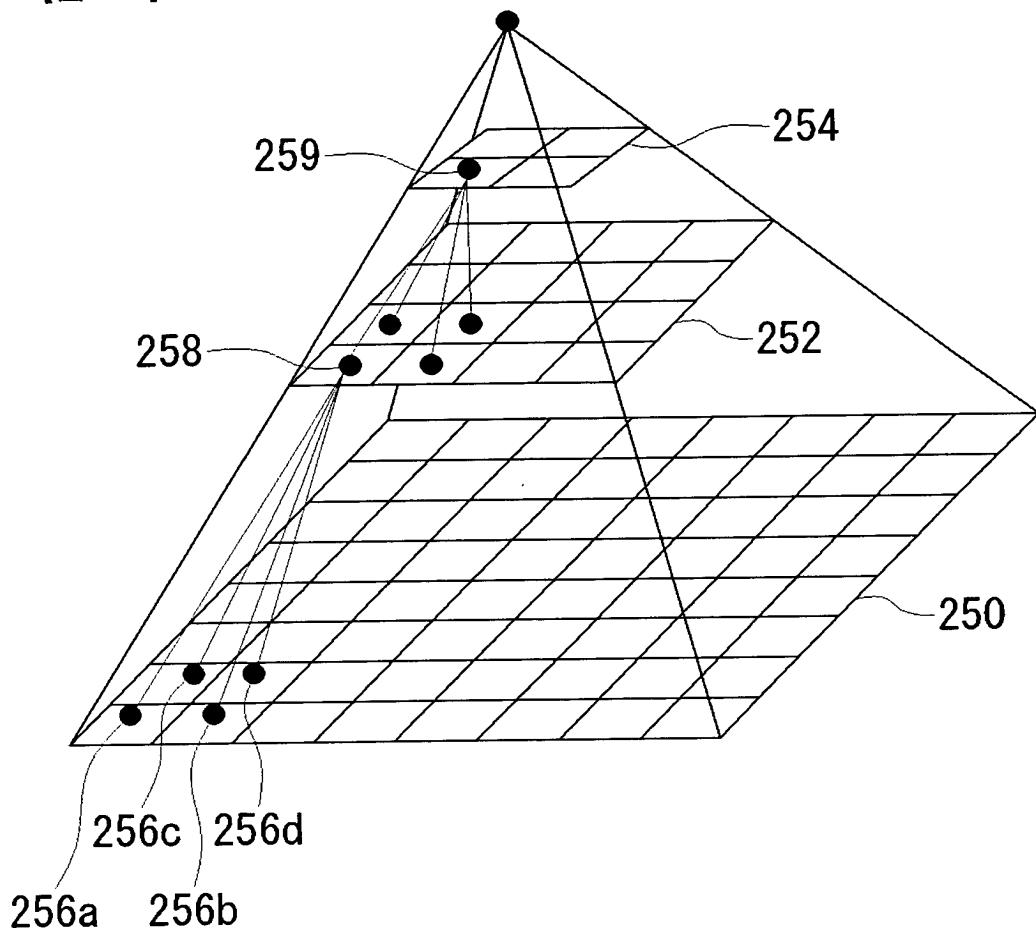


(a)

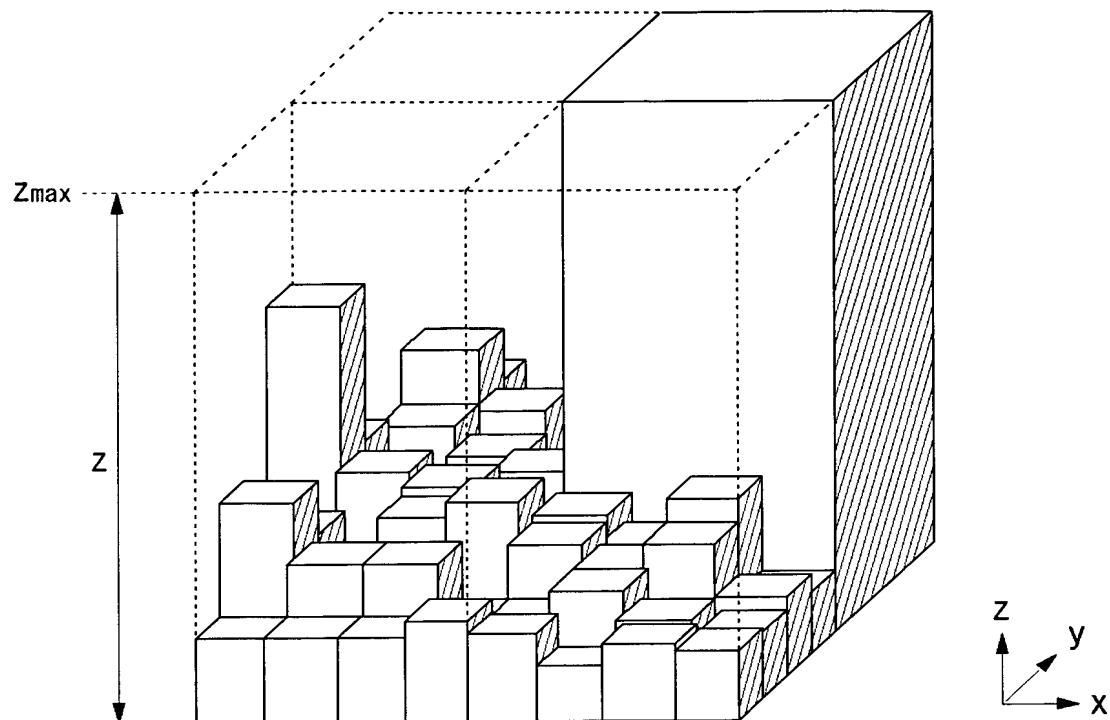


(b)

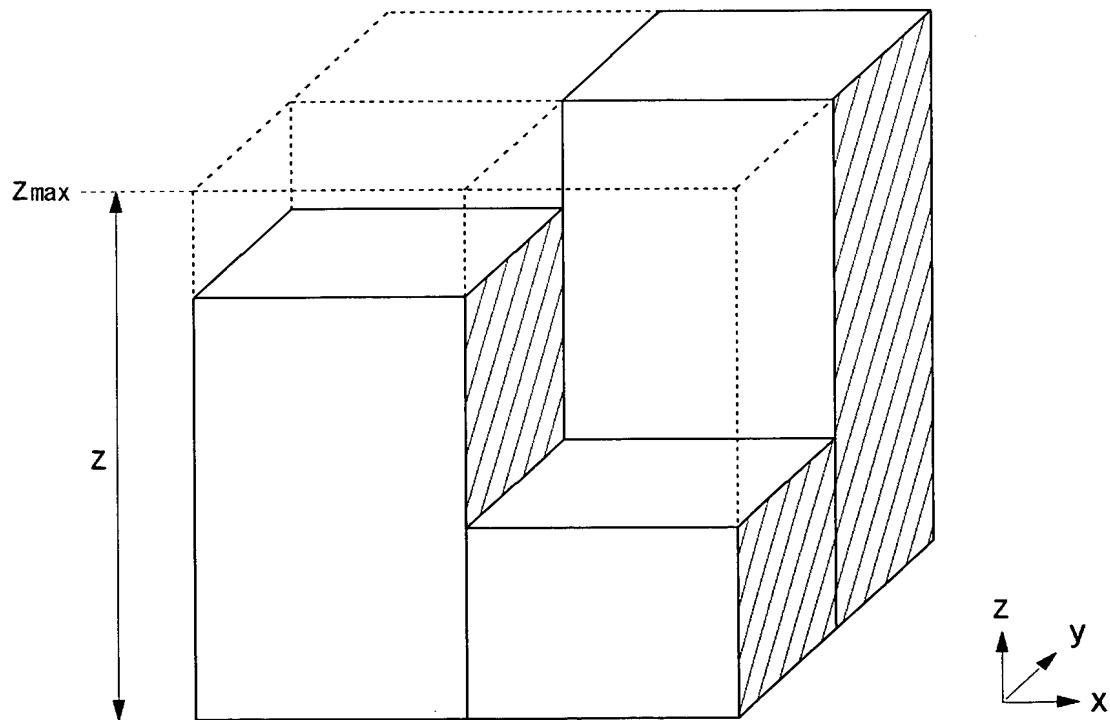
【図27】



【図28】

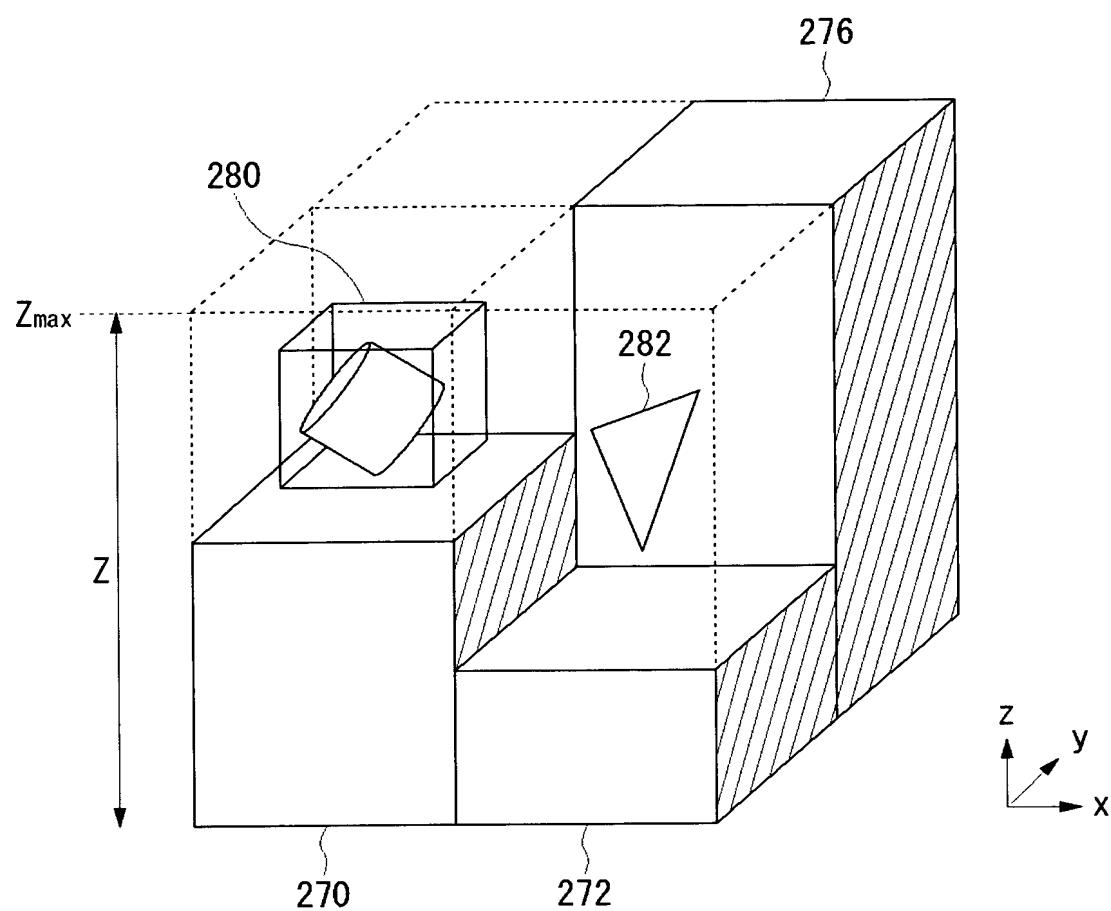
250

【図29】

254

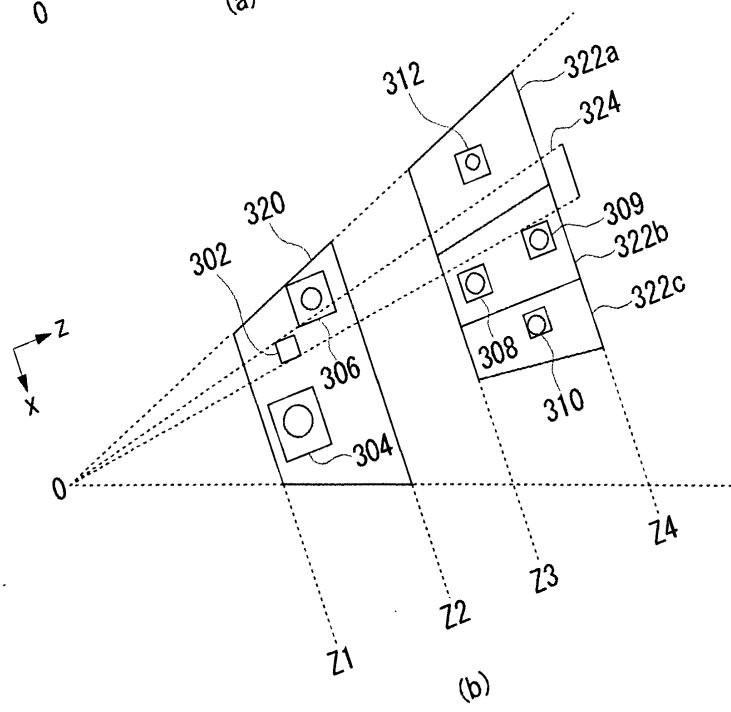
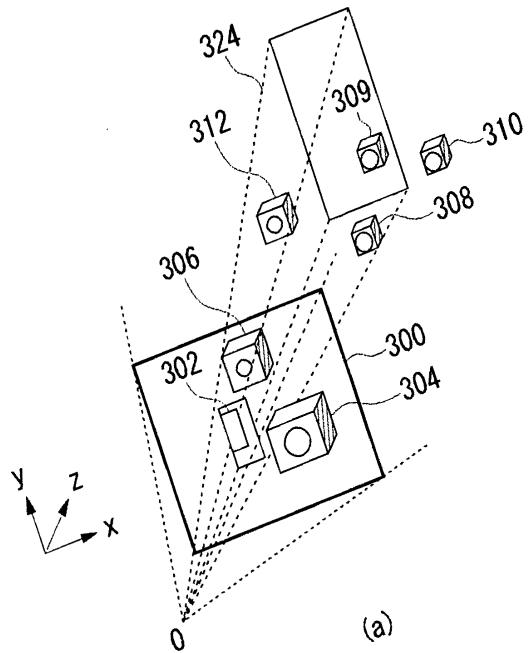
【図30】

290



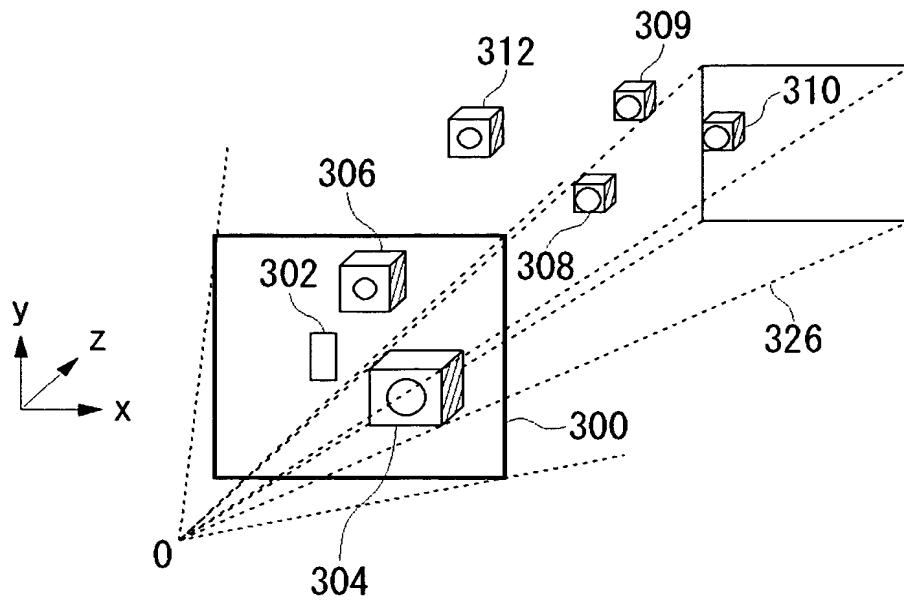
特願2003-293804

【図31】

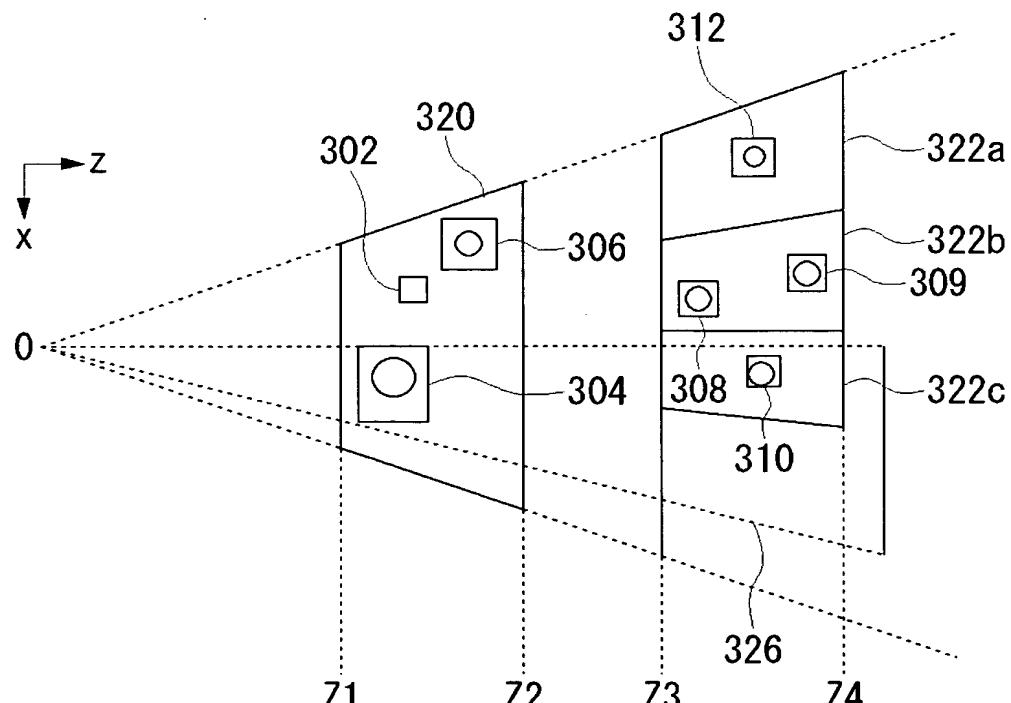


出証特2003-3070405

【図32】



(a)

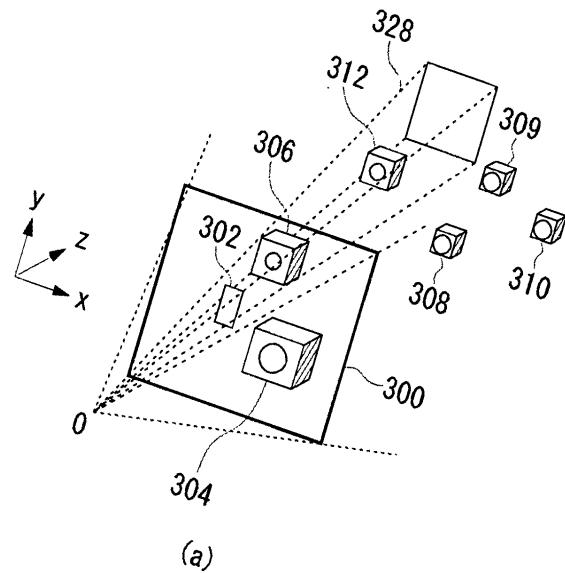


(b)

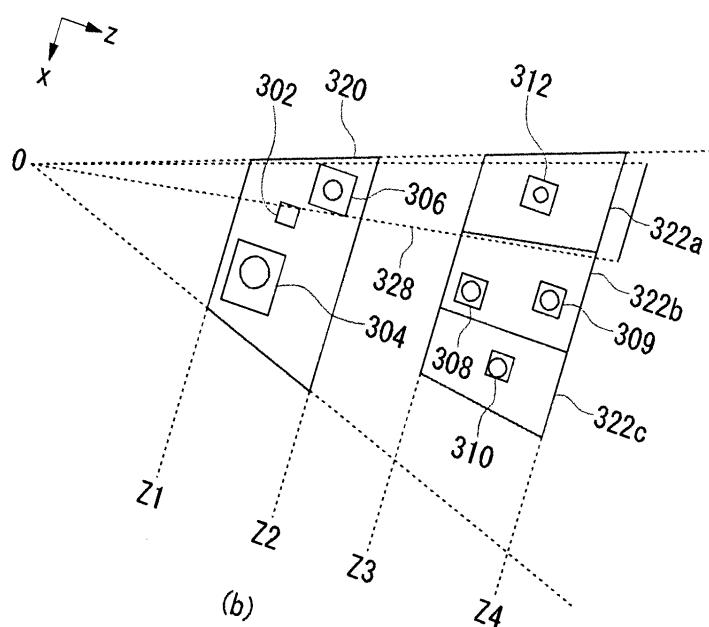
【图33】

特願 2003-293804

ページ： 31/E



(a)



(b)

出証特2003-3070405

【書類名】要約書

【要約】

【課題】 3次元オブジェクトのレンダリング処理を柔軟性や互換性を保ちながら、並列処理により高速化することは難しかった。

【解決手段】 オブジェクト記憶部52はオブジェクトの3次元データと、オブジェクトの占有する空間領域を包含するB-boxの配列データとを記憶する。分類部44はオブジェクト記憶部52からB-box配列を読み込み、オブジェクトの属性やLOD情報にもとづいてB-boxをグループに分類する。描画処理部46は、同一グループに属するB-boxを包含するブリックを算出し、ブリックごとに独立した描画処理を行って画像データを生成し、画像記憶部54に格納する。統合部48は、画像記憶部54に格納されたブリックごとの画像データを統合して最終的に表示すべき出力画像データを生成する。

【選択図】

図7

特願 2003-293804

▼  
出願人履歴情報

識別番号 [395015319]

1. 変更年月日 1997年 3月 31日  
[変更理由] 住所変更  
住 所 東京都港区赤坂7-1-1  
氏 名 株式会社ソニー・コンピュータエンタテインメント

2. 変更年月日 2003年 7月 1日  
[変更理由] 住所変更  
住 所 東京都港区南青山二丁目6番21号  
氏 名 株式会社ソニー・コンピュータエンタテインメント